A SPATIAL MULTIGRID ITERATIVE METHOD FOR TWO-DIMENSIONAL DISCRETE-ORDINATES TRANSPORT PROBLEMS

A Dissertation

by

BRIAN DAVID LANSRUD

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2005

Major Subject: Nuclear Engineering

A SPATIAL MULTIGRID ITERATIVE METHOD FOR

TWO-DIMENSIONAL DISCRETE-ORDINATES TRANSPORT PROBLEMS

A Dissertation

by

BRIAN DAVID LANSRUD

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

Marvin L. Adams (Chair of Committee)	Raytcho Lazarov (Member)
Ron R. Hart (Member)	Paul Nelson (Member)
William E. Burchill (Head of Department)	

May 2005

Major Subject: Nuclear Engineering

ABSTRACT

A Spatial Multigrid Iterative Method for Two-Dimensional
Discrete-Ordinates Transport Problems. (May 2005)
Brian David Lansrud, B.S., Texas A&M University;
B.S., Texas A&M University
Chair of Advisory Committee: Dr. Marvin L. Adams

Iterative solutions of the Boltzmann transport equation are computationally intensive. Spatial multigrid methods have led to efficient iterative algorithms for solving a variety of partial differential equations; thus, it is natural to explore their application to transport equations. Manteuffel *et al*. conducted such an exploration in one spatial dimension, using two-cell inversions as the *relaxation* or *smoothing* operation, and reported excellent results. In this dissertation we extensively test Manteuffel's one-dimensional method and our modified versions thereof. We demonstrate that the performance of such spatial multigrid methods can degrade significantly given strong heterogeneities. We also extend Manteuffel's basic approach to two-dimensional problems, employing four-cell inversions for the *relaxation* operation. We find that for uniform homogeneous problems the two-dimensional multigrid method is not as rapidly convergent as the one-dimensional method. For strongly heterogeneous problems the performance of the two-dimensional method is much like that of the one-dimensional method, which means it can be slow to converge. We conclude that this approach to spatial multigrid produces a method that converges rapidly for many problems but not for others. That is, this spatial multigrid method is not unconditionally rapidly convergent. However, our analysis of the distribution of eigenvalues of the iteration operators indicates that this spatial multigrid method may work very well as a preconditioner within a Krylov iteration algorithm, because its eigenvalues tend to be relatively well clustered. Further exploration of this promising result appears to be a fruitful area of further research.

DEDICATION

This dissertation, all work leading to it, all work that may be derived from it and any benefit that may come to the author as a result of it is dedicated to my father.

DAVID ENDRE LANSRUD
March 13, 1942 – July 27, 2004

"Not to laugh, not to lament, not to curse, but to understand." – Spinoza

ACKNOWLEDGEMENTS

> But [Brian], do you see, was a creature in the transition state – neither caterpillar nor butterfly. He was just enough civilized to show off his outlandishness in the strangest possible manner. His education was not yet completed. He was an undergraduate.
>
> -adapted from Herman Melville's *Moby Dick*

Similar to a consuming narcotics addiction, the pursuit of a Ph.D. is an enormously selfish endeavor that greedily consumes resources, financial and otherwise, that are often not the student's (*i.e.*, addict's) and serve to pacify only the student's immediate needs. I have been extraordinarily lucky to encounter a number of individuals that, despite my best efforts, have recognized some ability within me. To each of them, I owe an un-repayable debt. Because of them, Brian David Lansrud has been given the chance to have a more prosperous, fulfilling, meaningful and, most importantly, interesting life.

Prof. Marvin L. Adams has watched me evolve from a stubborn, ill-tempered undergraduate into a graduate student of tolerable adequacy. At every step in this long process, he has given me expert technical guidance coupled with patience and friendship. I hope our collaboration on this dissertation is only the first step in many fruitful years of tackling difficult problems together. I can think of no compliment great enough to capture the respect and admiration I have for him.

Prof. Ron R. Hart convinced me to attend graduate school. Over the years he has profoundly influenced my attitude toward academics and realizing personal potential. Prof. Hart is one of those few individuals who's influence will remain with me for the remainder of my days.

The final stage of my graduate career has been spent at Los Alamos National Laboratory. Two individuals are mainly responsible for my position at the laboratory and maintaining an environment that allowed me to concentrate on completing my dissertation – Drs. Don Burton and Bill Chandler. Bill has helped me learn that no matter

how difficult things seem they will always get better. Don has helped me learn that no matter how good things seem someone, somewhere is plotting against me. Having them as my mentors was an unparalleled privilege in an already privileged place.

In addition to these two gentlemen, I thank all of my friends at Los Alamos who have helped me along the way. A few who made a significant impact on this effort are: Mike Berry who deserves special recognition for giving generously of his time and expertise in parallel computing; Tom Gianakon, Jim Hill, Mack Kenamond, John Pedicini, Matt Kirkland, Kent Parson, Dave Becker, Langdon Bennett, and Bob Webster. While I have left some of them unnamed, they are not forgotten and certainly very appreciated.

To all of my friends I owe a great deal for keeping my spirits up during some very difficult times. It is very easy to get discouraged during a long academic career watching your best friends move on while you remain behind perpetually being asked the allegedly-supportive, yet subtly-hateful question "So…when will you be finished?" My friends have shown patience during the years of unreturned phone calls, unattended get-togethers, and forgotten significant dates. I am thankful for all of you, and, at the risk of making someone mad, I want to enumerate a few for their special role in my life – Michael K. West, Kyle C. Schurter, Seth D. Nielsen, John C. Hummel, James E. Manuel, Paul B. Johns, Scott M. Johns, R. Cable Kurwitz, John A. Rennie, Steve Buck, and John D. Shipp.

It would be negligent of me to not recognize a special group of individuals who have certainly had their impact on my academic career. To all of my ex-girlfriends, of any duration, I thank each of you for helping me learn the pointlessness of maintaining effort without the prospect of reward.

There are a few individuals who have influenced me in unique ways. I recognize them without details – Dr. David A. Sherron, Dr. M. Theresa Quinn and President Richard M. Nixon.

Finally, I thank my mother and father – Mary Louise and David Endre Lansrud. It is only because of their love, patience, and generosity that their youngest had the

chance to even consider a Ph.D. They quietly watched a lot of goofing off before their son finally found something he considered worth the effort. The satisfaction of completing this work is tempered by the wish that I had worked harder so my father would have had the chance to enjoy it with me.

NOMENCLATURE

$\underline{\Psi}^{\ell}$ - angular flux density at grid level ℓ.

$\underline{\Psi}_i^{\ell}$ - angular flux density associated with cell *i*, at grid level ℓ - notation for the one-dimensional case.

$\underline{\Psi}_{K_{(ii,jj)}}^{\ell}$ - angular flux density associated with cell $K_{(ii,jj)}$, at grid level ℓ - notation for the two-dimensional case.

$\underline{\psi}^{\ell}$ - angular flux density *correction* at grid level ℓ.

$\underline{\psi}_i^{\ell}$ - angular flux density *correction* at grid level ℓ - notation for the one-dimensional case.

$\underline{\psi}_{L_i}^{+}$ - correction for the rightward-moving component of the angular flux density at the left edge of cell *i*. This is one piece of the cell-centered scalar flux correction.

$\underline{\psi}_{R_i}^{+}$ - correction for the rightward-moving component of the angular flux density at the right edge of cell *i*. This is the right edge exiting flux correction.

$\underline{\psi}_{L_i}^{-}$ - correction for the leftward-moving component of the angular flux density at the left edge of cell *i*. This is the left edge exiting flux correction.

$\underline{\psi}_{R_i}^{-}$ - correction for the leftward-moving component of the angular flux density at the right edge of cell *i*. This is one piece of the cell-centered scalar flux correction.

$\underline{\psi}_{K_{(ii,jj)}}^{\ell}$ - angular flux density *correction* at grid level ℓ - notation for the two-dimensional case.

$\underline{\psi}_{K_{(ii,jj)}^{kk}}^{\ell}$ - angular flux density *correction* associated with vertex *kk* of cell $K_{(ii,jj)}$ at grid level ℓ.

TABLE OF CONTENTS

CHAPTER Page

LIST OF FIGURES

Figure Page

LIST OF TABLES

Table Page

CHAPTER I

INTRODUCTION

Introduction to Chapter I

Boltzmann transport theory has found use in the simulation of many physical systems. Particle transport phenomena in nuclear reactors, stars, the atmosphere, fusion systems, radiation therapy systems and patients, and semiconductors are all modeled using Boltzmann's equation. Modern computational-transport research is driven by the need to efficiently solve these problems.

Two basic approaches to solving Boltzmann's equation are Monte Carlo methods and deterministic transport methods. Each approach has advantages and shortcomings. Monte Carlo methods are easier to apply to a wide range of problems and more readily facilitate complex geometries. Statistical errors associated with Monte Carlo methods, however, make it very difficult to obtain accurate solutions at multiple locations in the phase space of the problem or accurate estimates of the effects of small changes. Deterministic transport methods are better suited to obtaining solutions throughout phase space and to computing the effects of small changes in a given physical system. For problems well suited to simulation by deterministic transport, Monte Carlo typically displays poorer computational performance. Deterministic methods are usually more difficult to implement and can be more challenging to parallelize. Both approaches are useful, both need improvements to make them more widely useful, and both are subjects of much ongoing research and development.

The number of independent variables needed to completely describe the transport solution in physical systems of interest leads to systems of equations that are very large and thus difficult to invert directly. A steady-state three-dimensional problem with one hundred partitions in each spatial dimension, eight unknowns per spatial cell, twenty energy groups, and three hundred discrete directions has fifty billion unknowns. A time-dependent version of the same problem has this many unknowns per time step. There is

---

This dissertation follows the style and format of Journal of Computational Physics.

no method that will directly invert the resulting matrix of coefficients in any reasonable computational time and expense. Thus, iterative methods are used almost exclusively to solve deterministic transport problems of practical interest. In the present work we develop, analyze, and test a new iterative method for a certain class of deterministic transport methods.

The Discrete Transport Equation

The continuous Boltzmann transport equation is:

$$\frac{1}{v(E)}\frac{\partial\Psi(\vec{r},E,\underline{\Omega},t)}{\partial t}+\underline{\Omega}\cdot\vec{\nabla}\Psi(\vec{r},E,\underline{\Omega},t)+\sigma_t(\vec{r},E,t)\Psi(\vec{r},E,\underline{\Omega},t)= \int_{4\pi}\int_{0}^{\infty}\sigma_s(\vec{r},E'\to E,\underline{\Omega}'\to\underline{\Omega},t)\Psi(\vec{r},E',\underline{\Omega}',t)dE'd\Omega'+Q(\vec{r},E,\underline{\Omega},t). \quad (1.1)$$

This form of the Boltzmann's equation details two sources – scattering and external. While other sources could be detailed (e.g., fission or delayed particle emissions), we have chosen this presentation of Boltzmann's equation since this work focuses on the given sources. It can be shown that a solution to the continuous transport equation, Eq. (1.1), exists and is unique [1]; however, finding this solution presents an enormous challenge. Since analytic solutions are typically obtainable for only simplistic physical problems, in practical applications the continuous transport equation is approximated by a system of discrete equations.

The first variable to address in a general, discrete transport solution algorithm is the time variable. Integrating Eq. (1.1) over a time step from an initial time t_{i-1} to t_i and dividing by the width of the time interval we obtain Eq. (1.2):

$$\frac{\Psi(\vec{r},E,\underline{\Omega},t_i)-\Psi(\vec{r},E,\underline{\Omega},t_0)}{v(E)(t_i-t_{i-1})}+\underline{\Omega}\cdot\vec{\nabla}\Psi(\vec{r},E,\underline{\Omega},t_i)+\sigma_t(\vec{r},E,t_i)\Psi(\vec{r},E,\underline{\Omega},t_i)= \int_{4\pi}\int_{0}^{\infty}\sigma_s(\vec{r},E'\to E,\underline{\Omega}'\to\underline{\Omega},t_i)\Psi(\vec{r},E',\underline{\Omega}',t_i)dE'd\Omega'+Q(\vec{r},E,\underline{\Omega},t_i). \quad (1.2)$$

In Eq. (1.2) the time-step-averaged angular flux has been approximated to be the angular flux at the end of the time step – this is the *fully implicit* or *backward Euler*

approximation. The discretized time-dependent term is now used to augment the collision-rate-density and fixed-source terms:

$$\underline{\Omega}\cdot\vec{\nabla}\Psi\left(\vec{r},E,\underline{\Omega},t_i\right)+\tilde{\sigma}_t\left(\vec{r},E,t_i\right)\Psi\left(\vec{r},E,\underline{\Omega},t_i\right)= \int_{4\pi}\int_{0}^{\infty}\sigma_s\left(\vec{r},E'\to E,\underline{\Omega}'\to\underline{\Omega},t_i\right)\Psi\left(\vec{r},E',\underline{\Omega}',t_i\right)dE'd\Omega'+\tilde{Q}\left(\vec{r},E,\underline{\Omega},t_i\right), \tag{1.3}$$

where the augmented terms in Eq. (1.3) are given by:

$$\tilde{\sigma}_t\left(\vec{r},E,t_i\right)=\sigma_t\left(\vec{r},E,t_i\right)+\frac{1}{v\left(E\right)\left(t_i-t_0\right)}, \tag{1.4}$$

and:

$$\tilde{Q}\left(\vec{r},E,\underline{\Omega},t_i\right)=Q\left(\vec{r},E,\underline{\Omega},t_i\right)+\frac{\Psi\left(\vec{r},E,\underline{\Omega},t_0\right)}{v\left(E\right)\left(t_i-t_0\right)}. \tag{1.5}$$

Equation (1.4) describes the augmented total cross-section, and Eq. (1.5) describes the fixed-source augmented by the initial condition. At a given time-step, material quantities (i.e., atom densities) are either known, or calculated. From these quantities, all interaction cross-sections are tabulated. With these quantities in hand, we have described a transport problem at a single time-step, and it represents one step in a sequence of time-dependent problems.

The transport equation in Eq. (1.3) is continuous in space, direction, and energy. The second variable addressed in a discrete transport solver is the energy variable. The traditional discretization technique for this dependency is the *multigroup* approximation. The underlying assumptions in this approximation are that *i*) the energy dependency of the solution is separable from the angular and spatial dependencies; and *ii*) the energy dependence of the solution is known and can therefore be used to generate very accurately averaged cross sections. These assumptions further reduce the transport equation to the mono-energetic form:

$$\underline{\Omega}\cdot\vec{\nabla}\Psi\left(\vec{r},\underline{\Omega}\right)+\tilde{\sigma}_t\left(\vec{r}\right)\Psi\left(\vec{r},\underline{\Omega}\right)=\int_{4\pi}\sigma_s\left(\vec{r},\underline{\Omega}'\to\underline{\Omega}\right)\Psi\left(\vec{r},\underline{\Omega}'\right)d\Omega'+\tilde{Q}\left(\vec{r},\underline{\Omega}\right); \tag{1.6}$$

where cross-sections have been averaged over each discrete energy interval. There are a variety of methods for obtaining these averages for neutron cross-sections and photon

opacities that will not be discussed here [2,3]. Equation (1.6) is one equation in a couple set of equations – each representing an energy group. Each group-dependent equation is couple to every other equation through group-to-group scattering.

The transport equation of Eq. (1.6), is continuous in space and angle. The next variable typically addressed in a discrete transport solver is the angular dependency. There are two widely used methods for discretizing the angular variable – discrete-ordinates (S_N), and spherical harmonics (P_N). The work in this dissertation uses the discrete-ordinates method, thus, we will not be discussing the aspects of the spherical-harmonics method. The discrete-ordinates method uses quadrature to approximate angular integrals. We leave the details of this discretization to Chapter II (one-dimension), and Chapter IV (two-dimensions). Rewriting Eq. (1.6) with the discrete-ordinates approximation we obtain:

$$\underline{\Omega}_k \cdot \vec{\nabla}\Psi\left(\vec{r},\underline{\Omega}_k\right) + \tilde{\sigma}_t\left(\vec{r}\right)\Psi\left(\vec{r},\underline{\Omega}_k\right) = \frac{\sigma_s\left(\vec{r}\right)}{4\pi}\sum_{k'} w_{k'}\Psi\left(\vec{r},\underline{\Omega}_{k'}\right) + \tilde{Q}\left(\vec{r},\underline{\Omega}_k\right), \tag{1.7}$$

where $\underline{\Omega}_k = \mu\vec{i} + \eta\vec{j} + \xi\vec{k}$. In Eq. (1.7) we have assumed isotropic scattering and assumed a quadrature set with quadrature weights, w_k, normalized to 4π. Rewriting Eq. (1.7) in Cartesian geometry and defining $\Psi_k = \Psi\left(\Omega_k\right)$, we obtain the steady-state, mono-energetic, discrete-ordinates, three-dimensional transport equation with isotropic scattering:

$$\begin{aligned}\mu\frac{d\Psi_k\left(x,y,z\right)}{dx} + \eta\frac{d\Psi_k\left(x,y,z\right)}{dy} + \xi\frac{d\Psi_k\left(x,y,z\right)}{dz} + \tilde{\sigma}_t\left(x,y,z\right)\Psi_k\left(x,y,z\right) = \\ \frac{\sigma_s\left(x,y,z\right)}{4\pi}\Theta\left(x,y,z\right) + \tilde{Q}_k\left(x,y,z\right),\end{aligned} \tag{1.8}$$

where the discrete scalar flux is given by $\Theta\left(x,y,z\right) = \sum_{k'} w_{k'}\Psi_{k'}\left(x,y,z\right)$.

The transport equation of Eq. (1.8) is discrete in time, energy and angle, but it also has three continuous spatial dependencies. Spatial discretization methods for discrete-ordinates (S_N) transport problems is an area of active research. The discretization methods used in this work are families of Linear-Discontinuous Finite

Element Methods (LDFEMs) in one dimension (Chapter II), and Bilinear-Discontinuous Finite Element Methods (BLDFEMs) in two dimensions (Chapter IV). These discretization will be presented in detail in their respective chapters. In the next section we discuss selected iterative methods for solving the discrete form of the Boltzmann transport equation presented in Eq. (1.8).

Examples of Current Iterative Transport Methods

A commonly used procedure in discrete-ordinates (S_N) problems is the transport *sweep*. Sweep-based algorithms employ a *loop* over angles. For each angle, the angular flux in computed throughout the entire spatial domain, beginning at incoming boundaries and ending at outgoing boundaries. All angles are coupled through the scattering term (*i.e.*, the discrete scalar flux of Eq. (1.8)); therefore, a sweep-based iterative scheme must loop over all angles prior to beginning the next inner iteration. Sweep-based methods are difficult to parallelize because a given cell cannot be calculated until its upstream neighbors have been calculated. However, modern techniques have shown some success in this area [4].

One of the simplest sweep-based iterative methods used to solve the Boltzmann transport equation is source iteration. To illustrate this method, consider the one-dimensional form of Eq. (1.8):

$$\mu \frac{d\Psi_k(x)}{dx} + \sigma_t(x)\Psi_k(x) = \frac{\sigma_s(x)}{4\pi}\Theta(x) + Q_k(x). \tag{1.9}$$

One complete source iteration for Eq. (1.9) can be written as follows:

$$\mu \frac{d\Psi_k^{(n+1)}(x)}{dx} + \sigma_t(x)\Psi_k^{(n+1)}(x) = \frac{\sigma_s(x)}{4\pi}\Theta^{(n)}(x) + Q_k(x),\ n \geq 0, \tag{1.10}$$

where the initial guess for the scalar flux, $\Theta^{(0)}$, is set to an arbitrary value. Source iteration, Eq. (1.10), is a simple algorithm to implement; however, a Fourier analysis of the convergence behavior shows how poorly it performs for optically thick problems dominated by scattering. The analysis of an infinite, homogeneous problem shows that

the spectral radius, ρ_{SI}, for source iteration is equivalent to the scattering ratio, c, of the material:

$$\rho_{SI} = c = \frac{\sigma_s}{\sigma_t}. \tag{1.11}$$

The spectral radius, Eq. (1.11), is the factor by which the error is reduced per iteration after a lot of iterations (when only the slowest-converging mode remains in the error); thus, as the material becomes dominated by scattering, source iteration exhibits very poor convergence performance. Poor convergence performance makes source iteration an unattractive independent iterative scheme for problems with high scattering ratios.

Source iteration can be useful as part of more advanced preconditioning, or synthetic acceleration, schemes [5]. It is common for preconditioned transport schemes to begin with one complete source iteration:

$$\mu \frac{d\Psi_k^{(n+1/2)}(x)}{dx} + \sigma_t(x)\Psi_k^{(n+1/2)}(x) = \frac{\sigma_s(x)}{4\pi}\Theta^{(n)}(x) + Q_k(x),\ n \geq 0; \tag{1.12}$$

however, rather that resulting in the next iteration, $\Psi_k^{(n+1)}(x)$, we obtain an intermediate iteration value, $\Psi_k^{(n+1/2)}(x)$. Subtracting Eq. (1.12) from Eq. (1.9), we obtain the equation for an additive correction factor:

$$\begin{aligned} \mu \frac{d\psi_k^{(n+1/2)}(x)}{dx} + \sigma_t(x)\psi_k^{(n+1/2)}(x) = \frac{\sigma_s(x)}{4\pi}\phi^{(n+1/2)}(x) + \\ \frac{\sigma_s(x)}{4\pi}\left[\Theta^{(n+1/2)}(x) - \Theta^{(n)}(x)\right], \end{aligned} \tag{1.13}$$

for $n \geq 0$, where $\psi_k^{(n+1/2)}(x) = \Psi_k(x) - \Psi_k^{(n+1/2)}(x)$, and $\phi^{(n+1/2)}(x) = \sum_k w_k \psi_k^{(n+1/2)}$; however, this problem is no easier to solve than our original transport problem in Eq. (1.9). The goal of a preconditioning scheme is to generate an intermediate iteration value that is a more accurate approximation to the transport solution than would be obtained by source iteration alone.

One example of a preconditioned, iterative transport method is Diffusion Synthetic Acceleration (DSA). DSA begins with one transport sweep, or source iteration, identical to Eq. (1.12) for an intermediate angular flux iteration. This intermediate iteration value is then used to compute a scalar flux:

$$\Theta^{(n+1/2)}(x) = \sum_{k'} w_{k'} \Psi_{k'}^{(n+1/2)}(x). \tag{1.14}$$

This intermediate scalar flux of Eq. (1.14) is then used as the scattering source in a diffusion equation for the scalar flux correction factor:

$$-\frac{d}{dx}\frac{1}{3\sigma_t(x)}\frac{d\phi^{(n+1/2)}(x)}{dx} + \sigma_a(x)\phi^{(n+1/2)}(x) = \sigma_s(x)\left[\Theta^{(n+1/2)}(x) - \Theta^{(n)}(x)\right]. \tag{1.15}$$

This is the diffusion approximation of the transport equation for the exact correction. The result of Eq. (1.15) is then used to compute the next scattering source, Eq. (1.16), to be substituted back into Eq. (1.12):

$$\Theta^{(n+1)}(x) = \Theta^{(n+1/2)}(x) + \phi^{(n+1)}(x). \tag{1.16}$$

The superiority of DSA over source iteration is made very clear by the Fourier analysis of the homogeneous, infinite medium problem [5]. The spectral radius for DSA, ρ_{DSA}, is given by Eq. (1.17):

$$\rho_{DSA} \le 0.2247c; \tag{1.17}$$

where *c* is the same scattering ratio as defined in Eq. (1.11).

A difficulty in using the DSA method is the need for consistent spatial discretizations in Eqs. (1.12) and (1.15). For complex geometries, a consistent discretization scheme may be difficult to obtain. More significantly, consistent diffusion discretizations may be so complicated that the diffusion step of each iteration becomes extremely computationally costly, thus reducing or eliminating the computational gain obtained by a lower number of iterations. This is an important requirement. For optically thick problems dominated by scattering, the convergence performance for an inconsistent DSA method can degrade, and the method can even become divergent [5]. This complication has lead researchers to develop preconditioning methods in which the

low-order approximation to the scalar flux correction factor is obtained by solving another transport problem. This family of preconditioning methods is called Transport Synthetic Acceleration (TSA).

Rather than solving the low-order diffusion problem in Eq. (1.15) for a correction factor, TSA solves a transport equation that is less computationally expensive than the original problem. This eliminates the complication of needing a consistent spatial discretization for both a transport and a diffusion problem. TSA methods are the subject of ongoing research. For illustration, we use an example given by Adams and Larsen of TSA that uses an S_2 problem as the low-order approximation to the correction factor [5]:

$$\mu\frac{d\psi_k^{(n+1/2)}(x)}{dx}+\left[\sigma_t(x)-\beta\sigma_s(x)\right]\psi_k^{(n+1/2)}(x)- \frac{(1-\beta)\sigma_s(x)}{2}\left[\psi_+^{(n+1/2)}(x)+\psi_-^{(n+1/2)}(x)\right]=\frac{\sigma_s(x)}{2}\left[\Theta^{(n+1/2)}(x)-\Theta^{(n)}(x)\right]. \tag{1.18}$$

One method for solving the low-order problem of Eq. (1.18) is source iteration. As has already been discussed, source iteration will perform poorly if the lower-order problem is optically thick and dominated by scattering; however, the low-order problem of Eq. (1.18) is, typically, not run until fully converged. The spectral radius of the TSA method strongly depends on number of iterations used for the low-order approximation (M), and the scaling factor, β. When the optimal values of M, and β, are used, TSA drastically reduces the expense of source iteration; however, these parameters are problem-dependent, and TSA can perform poorly for thick problems dominated by scattering – an effect of the low-order problem not being fully converged [5].

The Multigrid Philosophy

A concise presentation of the history and development of multigrid methods, and a review of the relevant literature, is given by Trottenberg, *et al.* [6]. The first investigations into the *efficiency* of multigrid algorithms were presented by Brandt in the 1970's. These efforts evolved into his key paper, *Multi-Level Adaptive Solutions to Boundary-Value Problems* [7]. In this paper, Brandt presents a *multigrid philosophy*,

guidance for developing efficient multigrid methods, and tools for evaluating convergence performance.

The *multigrid philosophy* begins with a recognition of a series of nested grids describing the same physical domain: G^1, G^2, …, G^N, where G^N is a grid coarse enough to make the solution of a given problem computationally inexpensive. Brandt presents these grids with a notion of mesh spacing; however, we present a variation of this notation that indicates the number-of-cells on a given grid relative to another; for example G^ℓ, $G^{2\ell}$, …, $G^{\left(2^N\right)\ell}$, where there are *N* grids describing the same physical domain, each grid having half the number of cells of the previous grid in the series – this is the notation used to describe the grids in our one-dimensional case. We recognize that this notation does not generalize to more complex coarsening strategies; however, for our work it does concisely describe the relative number of cells from one grid to another. The un-converged solution to a given discrete problem will have error that can be decomposed into modes. These modes will be of varying frequencies. Modes that are high-frequency relative to the fine grid can be attenuated with some error relaxation procedure – *e.g.*, single-cell inversion. Error modes that are low-frequency relative to the fine-grid are not significantly attenuated during the previous step. Multigrid methods attempt to effectively attenuate these modes by changing the grid, thus causing low-frequency modes on a fine grid to appear as high-frequency error modes on a suitably coarser grid. Error relaxation at any grid level should be performed as few times as possible leaving errors on coarser grids for another step. Multigrid methods have no unique structure. In this work, we advance, sequentially, from a fine grid to a coarse grid, and then return to the fine grid. This type of a multigrid method leads to the description of a $V\left(\nu_1,\nu_2\right)$ cycle. The method begins at the finest grid level, progresses *down* the *V*-cycle to the coarsest grid level, and returns *up* the *V*-cycle until the next iteration has been constructed on the finest grid (Figure 1.1). A thorough example of a three-level $V\left(\nu_1,\nu_2\right)$ cycle is given in the next section.

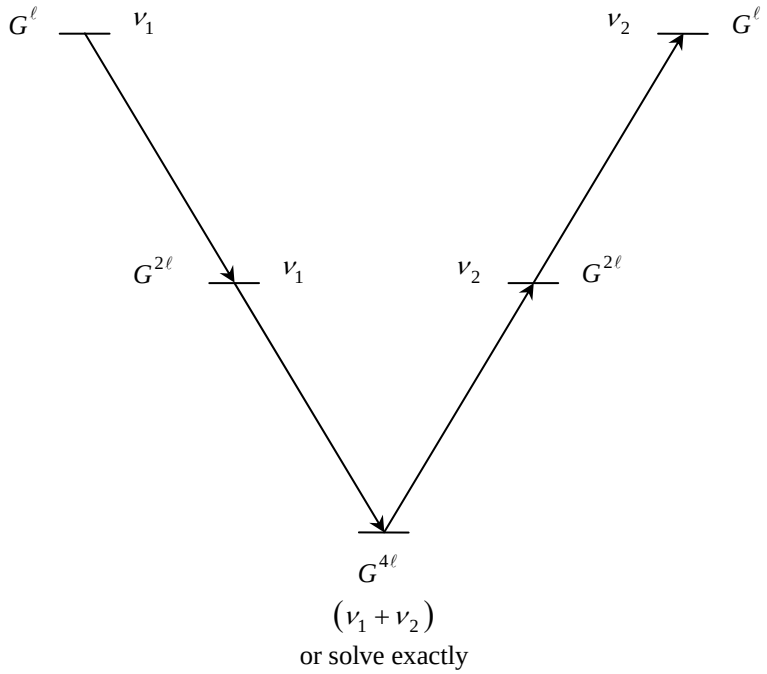

Figure 1.1 – Visualization of a Three-Level $V(\nu_1,\nu_2)$ Multigrid Cycle

A Spatial-Multigrid Tutorial

A spatial-multigrid method begins with a problem that can be written in matrix notation as:

$$A^{\ell}\underline{\Psi}^{\ell} = \underline{Q}^{\ell}, \tag{1.19}$$

where, in Eq. (1.19), A^{ℓ} is a matrix resulting from a system of equations, $\underline{\Psi}^{\ell}$ is the solution vector, and $\underline{Q}^{\ell}$ is the data, which comes from the boundary conditions and driving sources of the problem. A successful iterative method will yield the converged solution to the original problem. The matrix of coefficients, A^{ℓ}, can be written as the difference of two matrices. The converged solution satisfies:

$$A^{\ell}\underline{\Psi}^{\ell}_{(Conv)} = \left(L^{\ell} - R^{\ell}\right)\underline{\Psi}^{\ell}_{(Conv)} = \underline{Q}^{\ell}, \tag{1.20}$$

where in Eq. (1.20), ℓ represents the fine-grid partitioning. The choice of how to split the transport operator into $L^\ell - R^\ell$ should be made so that the following system is computationally inexpensive to solve:

$$L^\ell \underline{\Psi}^\ell_{(1)} = R^\ell \underline{\Psi}^\ell_{(0)} + \underline{Q}^\ell . \tag{1.21}$$

where $\underline{\Psi}^\ell_{(0)}$ is an initial approximation to the solution. The iterative method implied by Eq. (1.21) could be continued until the solution converges.

Until convergence is reached, each iterate contains some iteration error. One measure of that error is the "residual," which is defined as:

$$r^\ell_{(n)} = \underline{Q}^\ell - \left(L^\ell - R^\ell\right)\underline{\Psi}^\ell_{(n)} . \tag{1.22}$$

The residual defined by Eq. (1.22) often enters into the calculation in a multigrid iteration.

The error after a given number of iterations can be decomposed into modes. Error modes that vary on the scale of one cell (high-frequency error modes) will be rapidly attenuated by the iteration of Eq. (1.21) if the operator splitting is chosen properly [7]. When the splitting is so chosen, one iteration of Eq. (1.21) is called a *smoothing* or *relaxation* step. Error modes that vary slowly on the scale of one cell (low-frequency error modes) are more difficult to attenuate with a splitting that yields an easily solved Eq. (1.21). These low-frequency error modes, however, should be well resolved by a coarser grid. This is the motivation for multigrid methods, which approximately solve for the error by using successively coarser grids. On each grid, iterations of the form of Eq. (1.21) are performed to attenuate errors that are high-frequency *on that grid*.

For problems addressed by this research, the choices of L^ℓ and R^ℓ have physical interpretations. L^ℓ represents transport within a group of contiguous cells, and R^ℓ represents the connection between that group of contiguous cells and other such groups at grid level ℓ. Methods for constructing L^ℓ and R^ℓ matrices on coarser grids are the subject of active research, and there are numerous considerations influencing their

selection [7]. Their purpose, however, is the same – to effectively attenuate high-frequency error fluctuations at a given grid level.

A complete multigrid cycle estimates the amount by which the previous iterate needs to be corrected and uses this estimate to create the new iterate. Thus, after $n-1$ iterations, the n^{th} iterate is given by Eq. (1.23):

$$\underline{\Psi}^{\ell}_{(n)} = \underline{\Psi}^{\ell}_{(n-1)} + \underline{\psi}^{\ell}_{(n)}, \tag{1.23}$$

where $\underline{\psi}^{\ell}_{(n)}$ is the estimated additive correction. Ultimately, we want to compute a correction that will yield an n^{th} iterate that is close to the converged solution. Finding the exact correction would amount to solving the transport equation with a residual as the source term, as we now show. First compute the action of the fine-grid operator on the current iterate, then subtract it from both sides of Eq. (1.20) obtaining Eq. (1.24):

$$\left(L^{\ell} - R^{\ell}\right)\underline{\Psi}^{\ell}_{(Conv)} - \left(L^{\ell} - R^{\ell}\right)\underline{\Psi}^{\ell}_{(n)} = \underline{Q}^{\ell} - \left(L^{\ell} - R^{\ell}\right)\underline{\Psi}^{\ell}_{(n)}, \tag{1.24}$$

Define the correction corresponding to the n^{th} iterate on the fine mesh:

$$\underline{\varphi}^{\ell}_{(n)} = \underline{\Psi}^{\ell}_{(Conv)} - \underline{\Psi}^{\ell}_{(n)}, \tag{1.25}$$

Rewrite the right hand side of Eq. (1.24) in terms of Eq. (1.22), and substitute the definition of Eq. (1.25):

$$\left(L^{\ell} - R^{\ell}\right)\underline{\varphi}^{\ell}_{(n)} = r^{\ell}_{(n)}. \tag{1.26}$$

Eq. (1.26) shows that the n^{th} additive correction satisfies the original problem, Eq. (1.20), with the n^{th} residual as the source term. Of course, this additive correction cannot be explicitly computed. If it could, the problem would be solved in one step. The multigrid cycle approximately solves coarse-grid versions of Eq. (1.26) to obtain an estimate of this correction.

Recall that the operator splitting in Eq. (1.21) was chosen to *smooth* errors on the fine grid; that is, to attenuate error modes that are high-frequency relative to that grid. After a few iterations as in Eq. (1.21), the solution still contains low-frequency errors. The residual is computed as in Eq. (1.24), which after some algebra yields Eq. (1.27):

$$r^{\ell}_{(n+1/2)} = r^{\ell}_{(n)} - \left(L^{\ell} - R^{\ell}\right)\underline{\psi}^{\ell}_{(n+1/3)}, \tag{1.27}$$

where $\underline{\psi}^{\ell}_{(n+1/3)}$ represents the difference in the fine-grid approximation before and after relaxation, i.e., $\underline{\psi}^{\ell}_{(n+1/3)} = \underline{\Psi}^{\ell}_{(n+1/3)} - \underline{\Psi}^{\ell}_{(n)}$ (the subscript notation will be made clear in the example below). This residual is then used to drive a problem on a coarser grid, a problem whose solution will approximately correct for the error in the latest iterate.

The coarse-grid transport calculation to be performed requires operator splitting as described previously:

$$\left(L^{2\ell} - R^{2\ell}\right)\underline{\psi}^{2\ell}_{(n+1/3)} = r^{2\ell}_{(n)}, \tag{1.28}$$

where 2ℓ denotes the next coarser grid than grid level ℓ (note that Eq. (1.28) is a coarse-grid analog to the operation defined in Eq. (1.26)). For example, grid level 2ℓ will have half as many cells as grid level ℓ in a one-dimensional problem. Notation for the two-dimensional problem will be slightly different but will have the same interpretation. The source term for this equation is obtained by transferring the residuals computed in Eq. (1.27) from grid-level ℓ to 2ℓ with a restriction operator:

$$r^{2\ell}_{(n)} = I^{2\ell}_{\ell} r^{\ell}_{(n+1/2)}. \tag{1.29}$$

The restriction operator defined in Eq. (1.29), $I^{2\ell}_{\ell}$, selects information that will be used as boundary data for the coarser problem. The vector resulting from restriction is shorter in length than the original. High frequency residuals on grid 2ℓ are now attenuated in the same manner as those on grid level ℓ, i.e., by smoothing steps like Eq. (1.21) but on a coarser grid.

After smoothing on grid 2ℓ, a residual is computed on this grid and transferred to the next coarsest grid $\left(4\ell\right)$. The process continues recursively until the coarsest grid is reached. Following this procedure, each finer-grid solution is modified by transferring coarse-grid information and adding. As an example, the reconstructed fine-grid correction is obtained by the following:

$$\underline{\psi}^{\ell}_{(n+2/3)} \leftarrow \underline{\psi}^{\ell}_{(n+1/3)} + I^{\ell}_{2\ell}\underline{\psi}^{2\ell}_{(n+1/3)}, \tag{1.30}$$

where the prolongation operator in Eq. (1.30), $I^{\ell}_{2\ell}$, interpolates the fine-grid solution from coarser-grid information. The result of prolongation is a vector longer than the original. This correction and replacement is followed on each grid level by an additional smoothing step.

To illustrate this method in more detail, consider an eight-cell, one-dimensional transport problem. The converged solution to this problem satisfies Eq. (1.20), where the right-hand side now represents volumetric sources and incident fluxes. The data that is sent to the top level of the multigrid cycle is the latest computed estimate of the fine-mesh solution, $\underline{\Psi}_{(n)}$. The next estimate, $\underline{\Psi}_{(n+1)}$, is computed upon completion of one $V(\nu_1,\nu_2)$ cycle.

- **Input** : The latest guess to the fine grid solution $\underline{\Psi}_{(n)}$;
  - Compute $r^{\ell}_{(n)} = \underline{Q} - \left(L^{\ell} - R^{\ell}\right)\underline{\Psi}_{(n)}$;
  - Set $\underline{\Psi}_{(n+1/3,0)} = \underline{\Psi}_{(n)}$;
  - Relax ν_1 times on the eight-cell problem: $L^{\ell}\underline{\Psi}_{(n+1/3,m)} = \underline{Q} + R^{\ell}\underline{\Psi}_{(n+1/3,m-1)}$, where $1 \le m \le \nu_1$; define $\underline{\Psi}_{(n+1/3)} = \underline{\Psi}_{(n+1/3,\nu_1)}$;
  - Calculate an eight-cell correction: $\underline{\psi}^{\ell}_{(n+1/3)} = \underline{\Psi}_{(n+1/3)} - \underline{\Psi}_{(n)}$;
  - Compute the relaxed eight-cell residual: $r^{\ell}_{(n+1/2)} = r^{\ell}_{(n)} - \left(L^{\ell} - R^{\ell}\right)\underline{\psi}^{\ell}_{(n+1/3)}$;
  - Transfer residual to a four-cell problem through a restriction operator: $r^{2\ell}_{(n)} = I^{2\ell}_{\ell} r^{\ell}_{(n+1/2)}$;
  - Use this as a source term in a four-cell transport problem: $L^{2\ell}\underline{\psi}^{2\ell}_{(n+1/3)} - R^{2\ell}\underline{\psi}^{2\ell}_{(n+1/3)} = r^{2\ell}_{(n)}$;
  - Set the initial approximation to the four-cell correction to zero: $\underline{\psi}^{2\ell}_{(n+1/3,0)} = 0$

- Relax ν_1 times on the four-cell problem: $L^{2\ell}\underline{\psi}^{2\ell}_{(n+1/3,m)} = r^{2\ell}_{(n)} + R^{2\ell}\underline{\psi}^{2\ell}_{(n+1/3,m-1)}$;
- Compute the relaxed four-cell residual: $r^{2\ell}_{(n+1/2)} = r^{2\ell}_{(n)} - \left(L^{2\ell} - R^{2\ell}\right)\underline{\psi}^{2\ell}_{(n+1/3)}$, where $\underline{\psi}^{2\ell}_{(n+1/3)} = \underline{\psi}^{2\ell}_{(n+1/3,\nu_1)}$;
- Transfer the residual to a two-cell problem through a restriction operator: $r^{4\ell}_{(n)} = I^{4\ell}_{2\ell} r^{2\ell}_{(n+1/2)}$;
- There are two options on how to handle the problem on the coarsest grid:
  1. Further relaxation:
     - Relax $\nu_1 + \nu_2$ times on $L^{4\ell}\underline{\psi}^{4\ell}_{(n+1/3,m)} = r^{4\ell}_{(n)} + R^{4\ell}\underline{\psi}^{4\ell}_{(n+1/3,m-1)}$, where $1 \le m \le \nu_1 + \nu_2$;
     - Set $\underline{\psi}^{4\ell}_{(n+1)} = \underline{\psi}^{4\ell}_{(n+1/3,\nu_1+\nu_2)}$
     - Note: $r^{4\ell}_{(n+1)} = r^{4\ell}_{(n)} - \left(L^{4\ell} - R^{4\ell}\right)\underline{\psi}^{4\ell}_{(n+1/3,\nu_1+\nu_2)}$;
  2. Solve exactly:
     - Directly invert the two-cell operator $\underline{\psi}^{4\ell}_{(n+1)} = \left(L^{4\ell} - R^{4\ell}\right)^{-1} r^{4\ell}_{(n)}$;
     - Note: $r^{4\ell}_{(n+1)} = 0$;
- Transfer the solution to the four-cell problem through a prolongation operator, and add to the previous four-cell correction. This will set appropriate boundary values and interpolate four-cell quantities: $\underline{\psi}^{2\ell}_{(n+2/3)} \leftarrow \underline{\psi}^{2\ell}_{(n+1/3)} + I^{2\ell}_{4\ell}\underline{\psi}^{4\ell}_{(n+1)}$;
- Relax ν_2 times on the four-cell problem: $L^{2\ell}\underline{\psi}^{2\ell}_{(n+1,m)} = R^{2\ell}\underline{\psi}^{2\ell}_{(n+1,m-1)} + r^{2\ell}_{(n)}$, where $\underline{\psi}^{2\ell}_{(n+1,0)} \equiv \underline{\psi}^{2\ell}_{(n+2/3)}$; Note: $r^{2\ell}_{(n+1)} = r^{2\ell}_{(n+1/2)} - \left(L^{2\ell} - R^{2\ell}\right)\underline{\psi}^{2\ell}_{(n+1)}$, where $\underline{\psi}^{2\ell}_{(n+1)} = \underline{\psi}^{2\ell}_{(n+1,\nu_2)}$;
- Transfer the solution to the eight-cell problem through a prolongation operator and add to the previous eight-cell correction: $\underline{\psi}^{\ell}_{(n+2/3)} \leftarrow \underline{\psi}^{\ell}_{(n+1/3)} + I^{\ell}_{2\ell}\underline{\psi}^{2\ell}_{(n+1)}$;

- Relax ν_2 times on the eight-cell problem $L^\ell \underline{\psi}^\ell_{(n+1,m)} = R^\ell \underline{\psi}^\ell_{(n+1,m-1)} + r^\ell_{(n)}$, where $\underline{\psi}^\ell_{(n+1,0)} = \underline{\psi}^\ell_{(n+2/3)}$; Note: $r^\ell_{(n+1)} = r^\ell_{(n+1/2)} - \left(L^\ell - R^\ell\right)\underline{\psi}^\ell_{(n+1)}$, where $\underline{\psi}^\ell_{(n+1)} = \underline{\psi}^\ell_{(n+1,\nu_2)}$;
- Compute the next iteration for the solution: $\underline{\Psi}_{(n+1)} = \underline{\Psi}_{(n)} + \underline{\psi}^\ell_{(n+1)}$, where $\underline{\psi}^\ell_{(n+1)} = \underline{\psi}^\ell_{(n+1,\nu_2)}$;

- **Output** : The next fine grid iterate, $\underline{\Psi}_{(n+1)}$.

Multigrid Iterative Methods for Second Order Partial Differential Equations

Partial differential equations of second order are often classified by three categories [8]. For the two-dimensional, second order problem in Eq. (1.31):

$$L\Psi = a_{11}\frac{\partial^2\Psi}{\partial x^2} + a_{12}\frac{\partial^2\Psi}{\partial x \partial y} + a_{22}\frac{\partial^2\Psi}{\partial y^2} + a_1\frac{\partial\Psi}{\partial x} + a_2\frac{\partial\Psi}{\partial y} + a_0\Psi\,, \tag{1.31}$$

L can be classified as:

- *elliptic* if $4a_{11}a_{22} > a_{12}^2$,
- *hyperbolic* if $4a_{11}a_{22} < a_{12}^2$,
- *parabolic* if $4a_{11}a_{22} = a_{12}^2$.

In their textbook *Multigrid*, Trottenberg, *et al.*, describe several successful multigrid techniques for solving problems described by Eq. (1.31) [6]. Elliptic problems are particularly well suited for multigrid solution methods. Trottenberg shows excellent convergence performance when solving Poisson's equation using a finite-difference spatial discretization, Gauss-Seidel error relaxation, full-weighting restriction and bilinear interpolation for prolongation. This model problem is treated with a range of cell-sizes and multigrid cycles, and converges with an L_2-norm estimate of the spectral radius in the range of $0.063 \le \rho \le 0.12$. Two features that make multigrid solution methods attractive for this model problem are the computationally inexpensive error

smoothing step, Gauss-Seidel, and the rapid convergence such that the solution gains, roughly, one significant digit per multigrid cycle. Trottenberg, *et al.*, go on to explore hyperbolic and parabolic problems using finite-difference and finite-volume spatial discretizations. This is relevant to the transport community since solutions to second-order forms of the transport equation can exhibit parabolic, hyperbolic, and elliptic properties. Further, finite-difference and finite-volume methods have been, and continue to be, used to discretize the discrete-ordinates equation.

Researchers solving fluid-flow problems have demonstrated the rapid performance of multigrid iterative schemes when solving problems described by the Navier-Stokes equations and the Euler equations. These problems can be challenging since they can exhibit characteristics of all three types, and combinations of, second order partial differential equations [6]. Despite the complexity, researchers have used multigrid to solve two and three-dimensional fluid-flow problems on structured and unstructured grids on serial and parallel computing frameworks [9-19]. This research experience has given discrete-ordinates transport researchers the motivation to explore the application of multigrid methods to their problems. Existing research is limited and is restricted to one and two-dimensional problems on structured, Cartesian grids.

Current Applications of Spatial-Multigrid Methods to S_N Transport Problems

The first published attempt to implement spatial multigrid iterative methods to discrete-ordinates transport problems was by Nowak, Larsen and Martin [20]. The smoothing step used by these authors, defined in Eq. (1.21), was simple source iteration (the terms relaxation and smoothing are used interchangeably - both refer to attenuation of error on a given grid). While the results were encouraging, the method was subject to the shortcoming of its relaxation method, which loses effectiveness as cells become optically thick. The same authors investigated spatial multigrid for two-dimensional transport problems with an improved relaxation method [21]. Again, the results were encouraging for simple discretization schemes but proved to be less effective for more accurate spatial discretizations. Barnett, Morel, and Harris used a relaxation method

combining source iteration and overlapping two-cell inversion [22]. They expanded on previous work by employing a linear discontinuous finite element method and permitting anisotropic scattering, but they considered only one-dimensional Cartesian (slab) geometry.

Manteuffel, *et al.*, significantly improved upon these efforts, at least for the slab-geometry problems they addressed [23-28]. The relaxation method they developed proved to be simpler and more effective than previous schemes. Further, their method eliminated the transport *sweep* which makes it easier to implement on parallel computers. Along with numerical experimentation, Manteuffel *et al.* incorporated analysis of their method in various physical limits. For the problems they tested in one dimension, their method is currently the fastest discrete-ordinates transport solver (has the lowest maximum spectral radius) of any published iterative method to our knowledge.

The smoother used was a series of two-cell inversions. Physically, this represents solving for all angular fluxes on a two-cell problem in terms of its boundary conditions. Manteuffel's construction decomposes the two-cell operator by using the Sherman-Morrison formula. Physically, this is solving for the scalar fluxes and then reconstructing the angular fluxes. Mathematically, this relaxation step for the two cells i and $i+1$ takes the following form (cell numbering is from left to right, *i.e.*, cell $i+1$ is on the right boundary of cell *i*):

$$\begin{bmatrix} \underline{\psi}^{+}_{L_i} \\ \underline{\psi}^{+}_{R_i} \\ \underline{\psi}^{-}_{L_i} \\ \underline{\psi}^{-}_{R_i} \\ \underline{\psi}^{+}_{L_{i+1}} \\ \underline{\psi}^{+}_{R_{i+1}} \\ \underline{\psi}^{-}_{L_{i+1}} \\ \underline{\psi}^{-}_{R_{i+1}} \end{bmatrix} = \begin{bmatrix} \Pi_i & -\Theta_i \\ -\Gamma_{i+1} & \Pi_{i+1} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 2\mathrm{M}_i \underline{\psi}^{+}_{L_i} \\ 0 \\ 0 \\ 0 \\ 2\mathrm{M}_{i+1} \underline{\psi}^{-}_{R_{i+1}} \\ 0 \end{bmatrix}, \tag{1.32}$$

where the diagonal blocks in Eq. (1.32) represent one-cell transport operators, and the off-diagonal blocks communicate interface angular fluxes between them; further, the subscripts L_i and R_i describe the location of the unknown – left or right edge of cell *i*, respectively; the superscripts, $\pm$, indicate rightward moving particles, +, or leftward moving particles, –. In the aforementioned algorithm, this is relaxation in a $V(1,1)$ multigrid cycle. The block 2×2 matrix shown is one block of the *L* matrix in the notation of the pervious subsection.

Manteuffel moves to coarser grids through a restriction operator that simply selects the boundary terms from each two-cell pair on the coarser grid. This type of restriction is termed an injection operator. Equation (1.33) represents an example of the restriction operator used by Manteuffel [27]:

$$I_{\ell}^{2\ell} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{1.33}$$

The prolongation operators used by Manteuffel take advantage of his discovery that the error across two cells, following a two-cell inversion, can be characterized by a linear function in certain interesting limits. For a purely scattering problem using an S_2 quadrature, Manteuffel shows that the error is exactly linear across each two-cell pair. Thus, fine-grid information can be interpolated, or prolonged, quite easily from coarse grid data [27]. For problems with absorption, Manteuffel uses a *kinked* linear interpolation method to account for the deviation from simple linearity he observed in the error following relaxation [28]. An example of the prolongation operator used by Manteuffel is given by Eq. (1.34):

$$I_{2\ell}^{\ell} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \left(\frac{1}{2}-d\right) & 0 & \left(\frac{1}{2}-d\right) & 0 \\ 0 & \left(\frac{1}{2}-d\right) & 0 & \left(\frac{1}{2}-d\right) \\ \left(\frac{1}{2}-d\right) & 0 & \left(\frac{1}{2}-d\right) & 0 \\ 0 & \left(\frac{1}{2}-d\right) & 0 & \left(\frac{1}{2}-d\right) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \tag{1.34}$$

where *d* is a measure of variation of his *kinked* functions from linearity [28].

The authors also analyzed the error following relaxation in various limits. This analysis showed that the methods of restriction and prolongation defined in Eqs. (1.33) and (1.34) would work well for problems other than those explicitly solved in their analysis, at least in the limit of either fine or coarse spatial cells. For example, the error in the fine mesh limit is a linear function up to $O\left(\max\left(\sigma_i^2 h_i^2, \sigma_{i+1}^2 h_{i+1}^2\right)\right)$. In problems that are optically thick and of varying absorption, the error after relaxation could still be characterized as piecewise linear up to $O(\sigma h)^{-1}$.

The results of numerical testing were even more encouraging. For the case of the fine-mesh limit, the error was $O\left(\max\left(\sigma_i^3 h_i^3, \sigma_{i+1}^3 h_{i+1}^3\right)\right)$; for the coarse grid, $O\left(\sigma^2 h^2\right)^{-1}$. Ultimately, none of the problems tested yielded an error-reduction factor, i.e., spectral radius, of greater than 0.078 – this corresponded to a homogeneous material partitioned with a non-uniform grid of random cell thicknesses.

Researches have recently investigated multigrid iterative techniques for solving multi-dimensional transport problems [29-31]. As previously mentioned, no known iterative method converges more rapidly than the one derived by Manteuffel, *et al*. These

results strongly encourage extension of this multigrid scheme to higher-dimensional problems.

Krylov Subspace Methods

Transport problems often result in operators that are neither symmetric nor positive definite. Problems of this type have been successfully accelerated using Krylov subspace methods [32]. These methods use the form of Eq. (1.35) for successive iteration values:

$$\Psi^{(n+1)} = \Psi^{(n)} + \psi^{(n)}, \tag{1.35}$$

where $\psi^{(n)}$ is selected from the Krylov space of dimension n defined by Eq. (1.36):

$$K_n\left(A, r^{(0)}\right) \equiv span\left\{r^{(0)}, Ar^{(0)}, A^2 r^{(0)}, \ldots, A^{n-1} r^{(0)}\right\}; \tag{1.36}$$

where:

$$r^{(n)} \equiv Q - A\Psi^{(n)}. \tag{1.37}$$

Note that Eq. (1.37) is the same residual calculation performed during a multigrid cycle. Similar to the conjugate gradient method (CG), the goal of a Krylov method is to select the *best* $\psi^{(n)}$ from the Krylov subspace K_n. A commonly used example of a Krylov subspace method is the *generalized minimum residual* (GMRES) method. For a more detailed discussion of these methods applied to transport problems, see the following references [5,32-34].

An application of these methods relevant to this work is a *preconditioned* Krylov subspace method. We know that Krylov methods will rapidly converge if the eigenvalues corresponding to the Fourier-decomposed eigenvectors are tightly clustered [33,34]. Precedent for this type of application exists for both diffusion synthetic acceleration (DSA) and transport synthetic acceleration (TSA). Researchers have shown that Krylov methods can be successful even when applied to iteration schemes that would otherwise be divergent [32].

In the event that our multigrid iterative methods are not effective independent solvers, we can analyze problem-dependent eigenvalue spectra. The clustering of the eigenvalues will give an indication if our multigrid methods could be used as preconditioners for Krylov subspace methods. This is an important option and one that will likely be needed, since no existing transport iterative method will rapidly converge across the entire domain of physical problems of interest.

Goals of this Work

In this dissertation, we will address a solution method that is fundamentally different from those described in Section III of this Chapter. We will not use transport *sweeps* – this eases parallelization concerns. Our method will involve a sequence of computations that will address residuals *on different grids* rather than approximating a correction with a low-order problem. It is also useful to comment on what this work will not address. This work uses existing discretization methods for spatial and angular variables and is subject to all of their known shortcomings – for example, the potential for negative fluxes and the existence of ray effects. When appropriate, we will highlight and discuss difficulties and concerns that are specific to this research but will be limited to this context.

The goal of this work is to expand the application of *spatial-multigrid methods* to solve two-dimensional, deterministic transport methods. The essentials of spatial-multigrid methods, and the existing efforts to apply them to deterministic transport problems, will be discussed in Chapter II. Chapter III will address the one-dimensional problem and highlight those factors affecting method performance. Chapter IV will present our extension of spatial-multigrid methods to two-dimensional deterministic transport problems. Chapter V will present numerical results of the two-dimensional implementation, factors affecting method performance, and options for mitigating poor convergence behavior. Chapter VI will summarize the major conclusions of this research and suggest future interesting work in this area.

As the discussion of DSA and TSA implies, there is no transport method that has been developed that exhibits superior computational performance over the entire domain of physical problems. This experience has lead researchers to investigate combinations of solution methods. Rather than viewing, for example, TSA as stand-alone solution scheme, it can be view as a very advanced preconditioner for a class of solvers known as *Krylov methods* [5]. A specific example of a Krylov method that has been successfully applied to deterministic transport problems is the *generalized minimum residual* (GMRES) method [32]. The details of this application are not the focus of this dissertation; however, the results of this work are enriched, and are more encouraging, when viewed in a similar context.

Summary of Chapter I

Developing fast and accurate solution methods to Boltzmann's equation is an active area of research. The complexity of the equation, when describing physically significant problems, makes analytic solutions virtually impossible to obtain. Numerical methods are the only reasonable option for complex problems. Computational methods require moving from the continuous Boltzmann equation to a set of discrete equations. The most active area of current research is in reducing the computational expense of the discrete *spatial* transport problem – often referred to as the *inner* iteration in a transport solver. Physically meaningful problems can be posed that will cause both deterministic and Monte Carlo solution techniques to have computationally unacceptable expense. Both research communities seek to develop faster methods – accelerators and preconditioners for deterministic methods, and variance-reduction techniques for Monte Carlo methods. Two transport accelerators discussed in this chapter have greatly improved upon simple source iteration; however, both have their shortcomings. Recent success with Krylov methods has shifted the view of these methods from independent solvers to advanced preconditioners. This work will present an extension of existing one-dimensional work on spatial-multigrid methods, and the development of a two-

dimensional spatial-multigrid method. When these methods fail to perform effectively, their preconditioning properties will be investigated.

Iterative error-reduction methods under a variety of names have been explored by researchers for nearly seventy years. Multigrid methods have evolved from earlier efforts culminating in the systematic investigation of their efficiency by Brandt and the codification of the multigrid philosophy. These methods have enjoyed a great deal of success when applied to elliptic problems, but the deterministic transport community has had limited experience and success with them. The one-dimensional S_N work of Manteuffel, *et al.*, displays the greatest success to date, and motivates the work in this dissertation to expand his methodology to two-dimensions.

A Guide To The Remaining Chapters

The following chapter, Chapter II, expands and explores the one-dimensional problem. The description of the multigrid method will be made precise through definition of the necessary operators; when possible, derivations will be given to predict method performance for simple cases.

Chapter III presents results from a Fourier error-mode analysis of the S_2, one-dimensional problem and compare them against observed convergence performance obtained with a one-dimensional, S_N, spatial-multigrid transport code – S_NAPPER_1D.

Chapter IV presents the two-dimensional, bilinear discontinuous finite element equations, in block-matrix form suitable for later multigrid manipulation. Chapter V presents a complete Fourier analysis of the S_2, two-dimensional problem with numerical results. It uses the portions of the analysis to characterize slowest-converging error modes and uses these modes to devise multigrid operators that we expect to work well for such modes. It also compares the Fourier-analysis predictions against observed convergence performance from a two-dimensional, S_N, spatial-multigrid transport code – S_NAPPER_2D.

Chapter VI summarizes the results and conclusions of this dissertation and suggests future work.

CHAPTER II

THE ONE-DIMENSIONAL PROBLEM

Introduction to Chapter II

Chapter I presents the discrete-ordinates (S_N) transport equation. This equation is discrete in time, energy and angular variables, but remains continuous in the spatial variable. The focus of this chapter is the application of the multigrid method for solving the one-dimensional, discrete-ordinates equation with isotropic scattering and isotropic fixed-source:

$$\mu_k \frac{d\Psi_k(x)}{dx} + \sigma(x)\Psi_k(x) = \frac{\sigma_s(x)}{2}\sum_{k'=1}^{N} w_{k'}\Psi_{k'}(x) + \frac{Q_k(x)}{2}. \qquad (2.1)$$

The spatial variable in Eq. (2.1) will be discretized using a family of Linear Discontinuous Finite Element Methods (LDFEMs) described in Section II. The basic unit of the multigrid method detailed in this chapter will be a contiguous, two-cell problem. To fully appreciate the details of this method in the context of LDFEMs, it helps to understand the spatial location and physical interpretations of the unknowns (see Figure 2.1 and defined Nomenclature).

While Figure 2.1 shows the location of the unknowns on the physical domain, it does not show the shape of the physical solution in the interior of the two-cell problem. The exact shape is not described by the LDFEMs, rather it is assumed to be linear (Figure 2.2). This unknown-location versus spatial-shape distinction is an important subtlety. During the multigrid cycle, spatially dependent angular flux residuals and corrections will have to be restricted and interpolated. Particular attention must be paid to how these operations are performed, what assumptions go into them, and what affect these assumptions have on method performance.

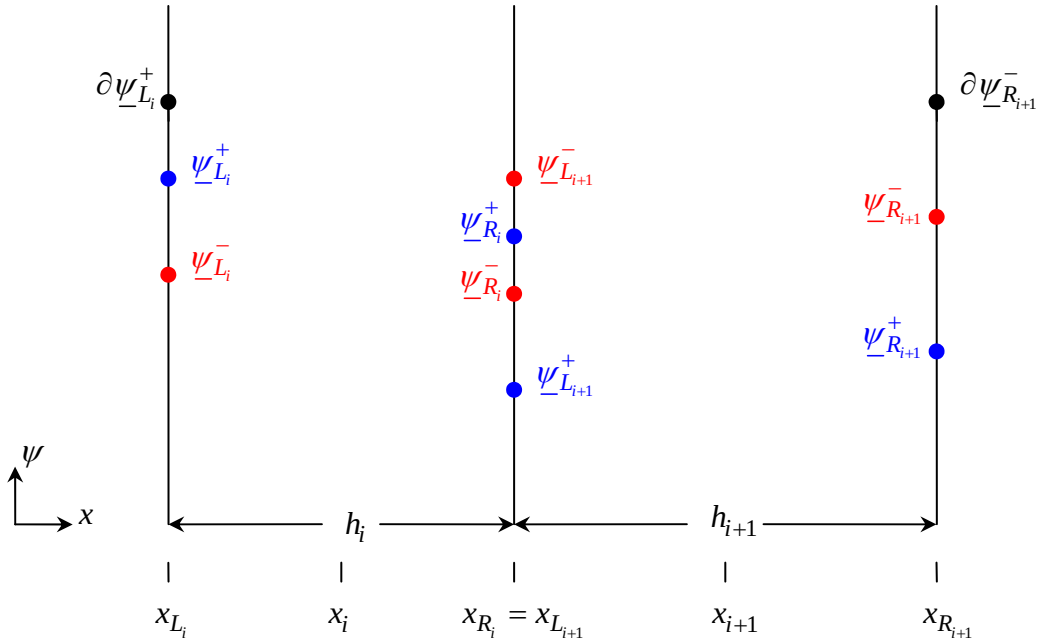

Figure 2.1 – Visualization of the LDFEM Unknowns

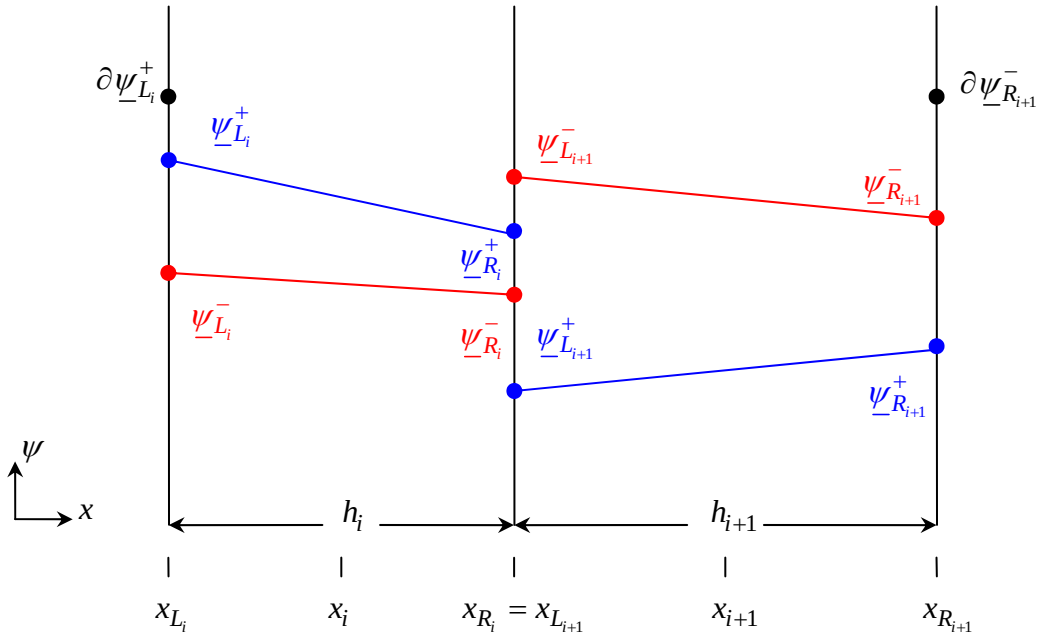

Figure 2.2 – Spatial Shape of the LDFEM Solution

One-Dimensional Linear Discontinuous Finite Element Methods (LDFEMs)

We choose to discretize the spatial variable using a family of linear discontinuous finite element methods (LDFEMs). Adams has shown that the family of LDFEMs preserves a number of desirable characteristics in problems with optically thick, highly-scattering cells – resolution, robustness, accuracy, and boundary conditions [35]. The robust characteristic is what leads to the excellent performance of Manteuffel's method in the thick-diffusive limit for one-dimensional problems.

The family of LDFEMs is a Galerkin method that can be described in terms of cardinal weight and basis functions. Cardinal functions are unity at a given support point and zero at all other support points. We are using a linear finite element method, thus, our weight and basis functions, defined in Eqs. (2.2) and (2.3), vary linearly from one to zero inside a given cell:

$$w_{L_i}(x) = b_{L_i}(x) = \frac{x_{R_i} - x}{h_i}, \tag{2.2}$$

$$w_{R_i}(x) = b_{R_i}(x) = \frac{x - x_{L_i}}{h_i}. \tag{2.3}$$

This choice of weight and basis functions results in a system of two equations and two unknown per angle per cell. To describe a family of LDFEMs, we present the equations with a mass-matrix lumping parameter, θ. As shown by Adams, lumping assures robustness in the thick-diffusive limit, but can reduce fine grid accuracy [35].

After multiplying Eq. (2.1) by the weight functions of Eqs. (2.2) and (2.3), and integrating over the volume of cell *i*, we obtain the standard LDFEM equations, which are given by $\theta_i = 3.0$ in the following equations, and the lumped LDFEM equations, which are given by $\theta_i = 1.0$ in the following equations:

$$\begin{aligned} &\mu_k\left(\Psi_{i,k} - \Psi_{i-1/2,k}\right) + \frac{\sigma_i h_i}{2}\left[\left(\frac{\theta_i + 1}{2}\right)\Psi_{L_i,k} + \left(\frac{\theta_i - 1}{2}\right)\Psi_{R_i,k}\right] = \\ &\frac{\sigma_{s,i} h_i}{4}\left[\left(\frac{\theta_i + 1}{2}\right)\Phi_{L_i} + \left(\frac{\theta_i - 1}{2}\right)\Phi_{R_i}\right] + \frac{1}{4}\left[\left(\frac{\theta_i + 1}{2}\right)Q_{L_i,k} + \left(\frac{\theta_i - 1}{2}\right)Q_{R_i,k}\right], \end{aligned} \tag{2.4}$$

$$\mu_k\left(\Psi_{i+1/2,k}-\Psi_{i,k}\right)+\frac{\sigma_i h_i}{2}\left[\left(\frac{\theta_i-1}{2}\right)\Psi_{L_i,k}+\left(\frac{\theta_i+1}{2}\right)\Psi_{R_i,k}\right]=$$
$$\frac{\sigma_{s,i}h_i}{4}\left[\left(\frac{\theta_i-1}{2}\right)\Phi_{L_i}+\left(\frac{\theta_i+1}{2}\right)\Phi_{R_i}\right]+\frac{1}{4}\left[\left(\frac{\theta_i-1}{2}\right)Q_{L_i,k}+\left(\frac{\theta_i+1}{2}\right)Q_{R_i,k}\right], \tag{2.5}$$

where:

$$\Phi_{H_i}=\sum_{k'=1}^{N} w_{k'}\Psi_{H_i,k'};\ H\in[L,R]. \tag{2.6}$$

Equations (2.4) and (2.5) describe the angular dependent particle flux density with angle, *k*, in a cell with total cross-section σ_i, scattering cross-section $\sigma_{s,i}$, lumping parameter θ_i, and width h_i, given the necessary boundary conditions. Further note that the quadrature weights in Eq. (2.6) have been normalized to two (2):

$$\sum_{k'=1}^{N} w_{k'}=2. \tag{2.7}$$

Traditional *sweep* based algorithms involve looping over each discrete ordinate, stepping through the problem domain one cell at a time for each ordinate. Our algorithm is not a *sweep* based method. In place of a transport *sweep*, we simultaneously solve for all discrete-ordinates in a given cell, or group of cells, including the scattering process. We facilitate this by writing Eqs. (2.4) - (2.5) in block-matrix form for a system of cells.

To demonstrate this construction, consider the S_2 form of Eqs. (2.4) and (2.5) for the angular-flux correction. As discussed in Chapter I, this corresponded to the homogeneous, or no fixed-source, problem:

$$\mu\left(\frac{\psi_{L_i,k}+\psi_{R_i,k}}{2}-\psi_{\partial L_i,1}\right)+\frac{\sigma_i h_i}{2}\left[\left(\frac{\theta_i+1}{2}\right)\psi_{L_i,k}+\left(\frac{\theta_i-1}{2}\right)\psi_{R_i,k}\right]=$$
$$\frac{\sigma_{s,i}h_i}{4}\left[\left(\frac{\theta_i+1}{2}\right)\left(\psi_{L_i,1}+\psi_{L_i,2}\right)+\left(\frac{\theta_i-1}{2}\right)\left(\psi_{R_i,1}+\psi_{R_i,2}\right)\right], \tag{2.8}$$

$$\mu\left(\psi_{R_i,1}-\frac{\psi_{L_i,1}+\psi_{R_i,1}}{2}\right)+\frac{\sigma_i h_i}{2}\left[\left(\frac{\theta_i-1}{2}\right)\psi_{L_i,1}+\left(\frac{\theta_i+1}{2}\right)\psi_{R_i,1}\right]=$$
$$\frac{\sigma_{s,i}h_i}{4}\left[\left(\frac{\theta_i-1}{2}\right)\left(\psi_{L_i,1}+\psi_{L_i,2}\right)+\left(\frac{\theta_i+1}{2}\right)\left(\psi_{R_i,1}+\psi_{R_i,2}\right)\right], \tag{2.9}$$

$$-\mu\left(\frac{\psi_{L_i,2}+\psi_{R_i,2}}{2}-\psi_{L_i,2}\right)+\frac{\sigma_i h_i}{2}\left[\left(\frac{\theta_i+1}{2}\right)\psi_{L_i,2}+\left(\frac{\theta_i-1}{2}\right)\psi_{R_i,2}\right]= \\ \frac{\sigma_{s,i}h_i}{4}\left[\left(\frac{\theta_i+1}{2}\right)\left(\psi_{L_i,1}+\psi_{L_i,2}\right)+\left(\frac{\theta_i-1}{2}\right)\left(\psi_{R_i,1}+\psi_{R_i,2}\right)\right], \tag{2.10}$$

$$-\mu\left(\psi_{\partial R_i,2}-\frac{\psi_{L_i,2}+\psi_{R_i,2}}{2}\right)+\frac{\sigma_i h_i}{2}\left[\left(\frac{\theta_i-1}{2}\right)\psi_{L_i,2}+\left(\frac{\theta_i+1}{2}\right)\psi_{R_i,2}\right]= \\ \frac{\sigma_{s,i}h_i}{4}\left[\left(\frac{\theta_i-1}{2}\right)\left(\psi_{L_i,1}+\psi_{L_i,2}\right)+\left(\frac{\theta_i+1}{2}\right)\left(\psi_{R_i,1}+\psi_{R_i,2}\right)\right]. \tag{2.11}$$

To write Eqs. (2.8) - (2.11) in matrix form, we define a data structure for the solution vector in a single cell *i*, with Eq. (2.12):

$$\underline{\psi}_i=\left[\psi_{L_i,1}\quad \psi_{R_i,1}\quad \psi_{L_i,2}\quad \psi_{R_i,2}\right]^T. \tag{2.12}$$

Eqs. (2.8) - (2.11) can now be written in matrix notation:

$$\left[\mathrm{N}+\sigma_i h_i \mathrm{M}_i-\frac{\sigma_{s,i}h_i}{2}S_i\right]\underline{\psi}_i=2\mu\underline{\psi}_{\partial i}, \tag{2.13}$$

where:

$$\mathrm{N}=\begin{bmatrix}\mu & \mu & 0 & 0\\ -\mu & \mu & 0 & 0\\ 0 & 0 & \mu & -\mu\\ 0 & 0 & \mu & \mu\end{bmatrix}, \tag{2.14}$$

$$\mathrm{M}_i=\begin{bmatrix}\left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right) & 0 & 0\\ \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right) & 0 & 0\\ 0 & 0 & \left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right)\\ 0 & 0 & \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right)\end{bmatrix}, \tag{2.15}$$

$$S_i = \begin{bmatrix} \left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right) \\ \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right) \\ \left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right) \\ \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right) & \left(\frac{2\theta_i-1}{2\theta_i}\right) & \left(\frac{2\theta_i+1}{2\theta_i}\right) \end{bmatrix}. \tag{2.16}$$

We now generalize the matrix notation in Eqs. (2.13) - (2.16) to describe an S_N problem with a further abstraction that will allow asymmetric quadrature sets (*p* points for the positive directions and *n* points for the negative directions) :

$$\left[\mathrm{N} + \sigma_i h_i \mathrm{M}_i - \frac{\sigma_{s,i} h_i}{2} S_i \right] \underline{\psi}_i = \underline{\psi}_{\partial i}, \tag{2.17}$$

where:

$$\mathrm{N} = \begin{bmatrix} \mathrm{N}^{(p\times p)} & \mathrm{N}^{(p\times p)} & 0 & 0 \\ -\mathrm{N}^{(p\times p)} & \mathrm{N}^{(p\times p)} & 0 & 0 \\ 0 & 0 & \mathrm{N}^{(n\times n)} & -\mathrm{N}^{(n\times n)} \\ 0 & 0 & \mathrm{N}^{(n\times n)} & \mathrm{N}^{(n\times n)} \end{bmatrix}, \tag{2.18}$$

$$\mathrm{M}_i = \begin{bmatrix} \mathrm{B}_i^{(p\times p)} & \mathrm{A}_i^{(p\times p)} & 0 & 0 \\ \mathrm{A}_i^{(p\times p)} & \mathrm{B}_i^{(p\times p)} & 0 & 0 \\ 0 & 0 & \mathrm{B}_i^{(n\times n)} & \mathrm{A}_i^{(n\times n)} \\ 0 & 0 & \mathrm{A}_i^{(n\times n)} & \mathrm{B}_i^{(n\times n)} \end{bmatrix}, \tag{2.19}$$

$$S_i = \begin{bmatrix} \mathrm{B}_i^{(p\times p)} R^{(p\times p)} & \mathrm{A}_i^{(p\times p)} R^{(p\times p)} & \mathrm{B}_i^{(p\times n)} R^{(n\times n)} & \mathrm{A}_i^{(p\times n)} R^{(n\times n)} \\ \mathrm{A}_i^{(p\times p)} R^{(p\times p)} & \mathrm{B}_i^{(p\times p)} R^{(p\times p)} & \mathrm{A}_i^{(p\times n)} R^{(n\times n)} & \mathrm{B}_i^{(p\times n)} R^{(n\times n)} \\ \mathrm{B}_i^{(n\times p)} R^{(p\times p)} & \mathrm{A}_i^{(n\times p)} R^{(p\times p)} & \mathrm{B}_i^{(n\times n)} R^{(n\times n)} & \mathrm{A}_i^{(n\times n)} R^{(n\times n)} \\ \mathrm{A}_i^{(n\times p)} R^{(p\times p)} & \mathrm{B}_i^{(n\times p)} R^{(p\times p)} & \mathrm{A}_i^{(n\times n)} R^{(n\times n)} & \mathrm{B}_i^{(n\times n)} R^{(n\times n)} \end{bmatrix}, \tag{2.20}$$

where:

$$\mathrm{A}_i^{(q\times r)} = \left(\frac{2\theta_i - 1}{2\theta_i}\right) I^{(q\times r)},\ \mathrm{B}_i^{(q\times r)} = \left(\frac{2\theta_i + 1}{2\theta_i}\right) I^{(q\times r)},\ q,r \in [n,p] \tag{2.21}$$

and, for constructing scalar fluxes:

$$R^{(q\times r)} = \underline{1}^{(q)} \underline{w}_{(r)}^{T}. \tag{2.22}$$

A final few operator definitions, together with Eqs. (2.17) - (2.22), give a compact set of notation for completely describing a discrete-ordinates problem. For an interior cell:

$$\Pi_i \underline{\psi}_i = \Gamma_{i-1} \underline{\psi}_{i-1} + \Theta_{i+1} \underline{\psi}_{i+1}, \tag{2.23}$$

where:

$$\Pi_i = \mathrm{N} + \sigma_i h_i \mathrm{M}_i - \left(\frac{\sigma_{s,i} h_i}{2}\right) S_i, \tag{2.24}$$

$$\Gamma_{i-1} = \begin{bmatrix} 0 & 2\mathrm{N}^{(p\times p)} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{2.25}$$

and,

$$\Theta_{i+1} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2\mathrm{N}^{(n\times n)} & 0 & 0 \end{bmatrix}. \tag{2.26}$$

With the notation of Eqs. (2.23) - (2.26), we now write the set of discrete LD equations in block-matrix that describe the angular flux correction for a system of γ cells:

$$\begin{bmatrix} \Pi_1^\ell & -\Theta_1^\ell & 0 & \cdots & \cdots & \cdots & 0 \\ -\Gamma_2^\ell & \Pi_2^\ell & -\Theta_2^\ell & & & & \vdots \\ 0 & -\Gamma_3^\ell & \Pi_3^\ell & -\Theta_3^\ell & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & -\Gamma_{\gamma-2}^\ell & \Pi_{\gamma-2}^\ell & -\Theta_{\gamma-2}^\ell & 0 \\ \vdots & & & & -\Gamma_{\gamma-1}^\ell & \Pi_{\gamma-1}^\ell & -\Theta_{\gamma-1}^\ell \\ 0 & \cdots & \cdots & \cdots & 0 & -\Gamma_\gamma^\ell & \Pi_\gamma^\ell \end{bmatrix} \begin{bmatrix} \underline{\psi}_1^\ell \\ \underline{\psi}_2^\ell \\ \underline{\psi}_3^\ell \\ \vdots \\ \underline{\psi}_{\gamma-2}^\ell \\ \underline{\psi}_{\gamma-1}^\ell \\ \underline{\psi}_\gamma^\ell \end{bmatrix} = \begin{bmatrix} \partial\underline{\psi}_1^\ell \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \partial\underline{\psi}_\gamma^\ell \end{bmatrix}. \qquad (2.27)$$

This notation introduces an indication of grid level. Eq. (2.27) describes the fine-grid problem denoted by the superscript ℓ. Recall that if grid-level ℓ has γ cells, then grid level 2ℓ has $\gamma/2$ cells. Eq. (2.27) can be written to describe the general problem for the angular flux rather that its correction. The right-hand-side of the equation would include the fixed source with the leftmost and rightmost cell-sources augmented by the boundary conditions.

The development of Eq. (2.27) follows the methodology of Manteuffel, *et al.*, but it describes a more general set of one-dimensional problems. Our form allows the analysis of the standard LD form in addition to the LLD form already examined by Manteuffel. In the next section we analyze the shape of the error following relaxation. The relaxation step is Manteuffel's two-cell inverse.

Manteuffel's One-Dimensional Smoothing Step

As discussed in Chapter I, the success of a multigrid method is determined by the selection of an error-smoothing step, and, following that step, develop a methodology to approximate the shape of the resulting error by a coarser grid. Manteuffel recognized that he could efficiently invert a two-cell transport operator. Using that inverse as a smoothing operator, he further recognized that the error following relaxation could be accurately characterized in several limits. Manteuffel considered transport problems with and without absorption in separate published reports [27,28]. In this section we discuss

Manteuffel's two-cell inversion methodology and his error characterization over the range of physically meaningful scattering ratios, $0 \le c \le 1$.

Consider a two-cell transport problem in the interior of the system described by Eq. (2.27). This problem in block-matrix form is given by:

$$\begin{bmatrix} \Pi_i^\ell & -\Theta_i^\ell \\ -\Gamma_{i+1}^\ell & \Pi_{i+1}^\ell \end{bmatrix} \begin{bmatrix} \underline{\psi}_i^\ell \\ \underline{\psi}_{i+1}^\ell \end{bmatrix} = \begin{bmatrix} \Gamma_{i-1} \underline{\psi}_{i-1}^\ell \\ \Theta_{i+2} \underline{\psi}_{i+2}^\ell \end{bmatrix}. \tag{2.28}$$

The two-cell operator in Eq. (2.28) is large, $\dim\left[\left(4p+4n\right)\times\left(4p+4n\right)\right]$, and sparse. To efficiently invert the two-cell operator, in $O\left(4p+4n\right)$ steps, Manteuffel recognized that it can be written as:

$$\begin{bmatrix} \Pi_i^\ell & -\Theta_i^\ell \\ -\Gamma_{i+1}^\ell & \Pi_{i+1}^\ell \end{bmatrix} = \Pi_0 - VW^T . \tag{2.29}$$

This decomposition described by Eq. (2.29) can be interpreted in terms of the discrete transport equation. The first term, Π_0, represents all streaming and collision-rate density information. The second term, VW^T, represents the scattering-rate density information. The decomposition of Eq. (2.29) allows the use of the Sherman-Morrison formula, Eq. (2.30), for computing the inverse:

$$\left(\Pi_0 - VW^T\right)^{-1} = \Pi_0^{-1} + \Pi_0^{-1} V \left(I - W^T \Pi_0^{-1} V\right)^{-1} W^T \Pi_0^{-1}. \tag{2.30}$$

For this inversion to be efficient, the streaming-plus-collision operator, Π_0, must be sparse and simple to invert; and, the scattering operator, VW^T, must not be of full rank. Both of these conditions are satisfied for a two-cell problem on the fine grid, given isotropic scattering. We show the coarse-grid problems satisfy those requirements in a later section.

Consider the streaming-plus-collision term in Eq. (2.24) for the two-cell problem defined in Eq. (2.29):

$$\Pi_0^{\ell} = \begin{bmatrix} A_i^{(p\times p)} & B_i^{(p\times p)} & 0 & 0 & 0 & 0 & 0 & 0 \\ C_i^{(p\times p)} & D_i^{(p\times p)} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & E_i^{(n\times n)} & F_i^{(n\times n)} & 0 & 0 & 0 & 0 \\ 0 & 0 & G_i^{(n\times n)} & H_i^{(n\times n)} & 0 & 0 & J_i^{(n\times n)} & 0 \\ 0 & K_{i+1}^{(p\times p)} & 0 & 0 & A_{i+1}^{(p\times p)} & B_{i+1}^{(p\times p)} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{i+1}^{(p\times p)} & D_{i+1}^{(p\times p)} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & E_{i+1}^{(n\times n)} & F_{i+1}^{(n\times n)} \\ 0 & 0 & 0 & 0 & 0 & 0 & G_{i+1}^{(n\times n)} & H_{i+1}^{(n\times n)} \end{bmatrix}, \quad (2.31)$$

where the individual blocks are defined in Eqs. (2.18) - (2.19), and Eqs. (2.25) - (2.26). Every block in Eq. (2.31) is a diagonal matrix. Physically, this means that all angles are decoupled – this is true when there is no scattering present. By construction, this matrix contains no scattering information. With some effort, it can be shown that the inverse of Eq. (2.31) can be written as:

$$\Pi_0^{-1} = \begin{bmatrix} Z_{11}^{(p\times p)} & Z_{12}^{(p\times p)} & 0 & 0 & 0 & 0 & 0 & 0 \\ Z_{21}^{(p\times p)} & Z_{22}^{(p\times p)} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{33}^{(n\times n)} & Z_{34}^{(n\times n)} & 0 & 0 & Z_{37}^{(n\times n)} & Z_{38}^{(n\times n)} \\ 0 & 0 & Z_{43}^{(n\times n)} & Z_{44}^{(n\times n)} & 0 & 0 & Z_{47}^{(n\times n)} & Z_{48}^{(n\times n)} \\ Z_{51}^{(p\times p)} & Z_{52}^{(p\times p)} & 0 & 0 & Z_{55}^{(p\times p)} & Z_{56}^{(p\times p)} & 0 & 0 \\ Z_{61}^{(p\times p)} & Z_{62}^{(p\times p)} & 0 & 0 & Z_{65}^{(p\times p)} & Z_{66}^{(p\times p)} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & Z_{77}^{(n\times n)} & Z_{78}^{(n\times n)} \\ 0 & 0 & 0 & 0 & 0 & 0 & Z_{87}^{(n\times n)} & Z_{88}^{(n\times n)} \end{bmatrix}, \quad (2.32)$$

where the individual blocks can be written in terms of the blocks in the original matrix:

$$\left[Z_{11}^{(p\times p)} \right]_{jj} = \left\{ \frac{\left[D_i^{(p\times p)} \right]_{jj}}{\left[A_i^{(p\times p)} \right]_{jj} \left[D_i^{(p\times p)} \right]_{jj} - \left[B_i^{(p\times p)} \right]_{jj} \left[C_i^{(p\times p)} \right]_{jj}} \right\}, \quad (2.33)$$

$$\left[Z_{21}^{(p\times p)}\right]_{jj} = -\left\{\frac{\left[C_i^{(p\times p)}\right]_{jj}}{\left[A_i^{(p\times p)}\right]_{jj}\left[D_i^{(p\times p)}\right]_{jj} - \left[B_i^{(p\times p)}\right]_{jj}\left[C_i^{(p\times p)}\right]_{jj}}\right\}, \tag{2.34}$$

$$\left[Z_{12}^{(p\times p)}\right]_{jj} = -\left\{\frac{\left[B_i^{(p\times p)}\right]_{jj}}{\left[A_i^{(p\times p)}\right]_{jj}\left[D_i^{(p\times p)}\right]_{jj} - \left[B_i^{(p\times p)}\right]_{jj}\left[C_i^{(p\times p)}\right]_{jj}}\right\}, \tag{2.35}$$

$$\left[Z_{22}^{(p\times p)}\right]_{jj} = \left\{\frac{\left[A_i^{(p\times p)}\right]_{jj}}{\left[A_i^{(p\times p)}\right]_{jj}\left[D_i^{(p\times p)}\right]_{jj} - \left[B_i^{(p\times p)}\right]_{jj}\left[C_i^{(p\times p)}\right]_{jj}}\right\}, \tag{2.36}$$

$$\left[Z_{51}^{(p\times p)}\right]_{jj} = \left[K_{i+1}^{(p\times p)}\right]_{jj}\left\{\frac{\left[C_i^{(p\times p)}\right]_{jj}}{\left[A_i^{(p\times p)}\right]_{jj}\left[D_i^{(p\times p)}\right]_{jj} - \left[B_i^{(p\times p)}\right]_{jj}\left[C_i^{(p\times p)}\right]_{jj}}\right\}\times \left\{\frac{\left[D_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj} - \left[B_{i+1}^{(p\times p)}\right]_{jj}\left[C_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.37}$$

$$\left[Z_{61}^{(p\times p)}\right]_{jj} = -\left[K_{i+1}^{(p\times p)}\right]_{jj}\left\{\frac{\left[C_i^{(p\times p)}\right]_{jj}}{\left[A_i^{(p\times p)}\right]_{jj}\left[D_i^{(p\times p)}\right]_{jj} - \left[B_i^{(p\times p)}\right]_{jj}\left[C_i^{(p\times p)}\right]_{jj}}\right\}\times \left\{\frac{\left[C_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj} - \left[B_{i+1}^{(p\times p)}\right]_{jj}\left[C_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.38}$$

$$\left[Z_{52}^{(p\times p)}\right]_{jj} = -\left[K_{i+1}^{(p\times p)}\right]_{jj}\left\{\frac{\left[A_i^{(p\times p)}\right]_{jj}}{\left[A_i^{(p\times p)}\right]_{jj}\left[D_i^{(p\times p)}\right]_{jj} - \left[C_i^{(p\times p)}\right]_{jj}\left[B_i^{(p\times p)}\right]_{jj}}\right\}\times \left\{\frac{\left[D_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj} - \left[B_{i+1}^{(p\times p)}\right]_{jj}\left[C_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.39}$$

$$\left[Z_{62}^{(p\times p)}\right]_{jj}=\left[K_{i+1}^{(p\times p)}\right]_{jj}\left\{\frac{\left[A_i^{(p\times p)}\right]_{jj}}{\left[A_i^{(p\times p)}\right]_{jj}\left[D_i^{(p\times p)}\right]_{jj}-\left[C_i^{(p\times p)}\right]_{jj}\left[B_i^{(p\times p)}\right]_{jj}}\right\}\times \left\{\frac{\left[C_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj}-\left[B_{i+1}^{(p\times p)}\right]_{jj}\left[C_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.40}$$

$$\left[Z_{33}^{(n\times n)}\right]_{kk}=-\left\{\frac{\left[H_i^{(n\times n)}\right]_{kk}}{\left[F_i^{(n\times n)}\right]_{kk}\left[G_i^{(n\times n)}\right]_{kk}-\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}}\right\}, \tag{2.41}$$

$$\left[Z_{43}^{(n\times n)}\right]_{kk}=\left\{\frac{\left[G_i^{(n\times n)}\right]_{kk}}{\left[F_i^{(n\times n)}\right]_{kk}\left[G_i^{(n\times n)}\right]_{kk}-\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}}\right\}, \tag{2.42}$$

$$\left[Z_{34}^{(n\times n)}\right]_{kk}=\left\{\frac{\left[F_i^{(n\times n)}\right]_{kk}}{\left[F_i^{(n\times n)}\right]_{kk}\left[G_i^{(n\times n)}\right]_{kk}-\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}}\right\}, \tag{2.43}$$

$$\left[Z_{44}^{(n\times n)}\right]_{kk}=-\left\{\frac{\left[E_i^{(n\times n)}\right]_{kk}}{\left[G_i^{(n\times n)}\right]_{kk}\left[F_i^{(n\times n)}\right]_{kk}-\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}}\right\}, \tag{2.44}$$

$$\left[Z_{55}^{(p\times p)}\right]_{jj}=\left\{\frac{\left[D_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj}-\left[B_{i+1}^{(p\times p)}\right]_{jj}\left[C_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.45}$$

$$\left[Z_{65}^{(p\times p)}\right]_{jj}=-\left\{\frac{\left[C_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj}-\left[B_{i+1}^{(p\times p)}\right]_{jj}\left[C_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.46}$$

$$\left[Z_{56}^{(p\times p)}\right]_{jj}=-\left\{\frac{\left[B_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj}-\left[C_{i+1}^{(p\times p)}\right]_{jj}\left[B_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.47}$$

$$\left[Z_{66}^{(p\times p)}\right]_{jj}=\left\{\frac{\left[A_{i+1}^{(p\times p)}\right]_{jj}}{\left[A_{i+1}^{(p\times p)}\right]_{jj}\left[D_{i+1}^{(p\times p)}\right]_{jj}-\left[C_{i+1}^{(p\times p)}\right]_{jj}\left[B_{i+1}^{(p\times p)}\right]_{jj}}\right\}, \tag{2.48}$$

$$\left[Z_{37}^{(n\times n)}\right]_{kk}=\left[J_i^{(n\times n)}\right]_{kk}\left\{\frac{\left[F_i^{(n\times n)}\right]_{kk}}{\left[F_i^{(n\times n)}\right]_{kk}\left[G_i^{(n\times n)}\right]_{kk}-\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}}\right\}\times \left\{\frac{\left[H_{i+1}^{(n\times n)}\right]_{kk}}{\left[F_{i+1}^{(n\times n)}\right]_{kk}\left[G_{i+1}^{(n\times n)}\right]_{kk}-\left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.49}$$

$$\left[Z_{47}^{(n\times n)}\right]_{kk}=-\left[J_i^{(n\times n)}\right]_{kk}\left\{\frac{\left[E_i^{(n\times n)}\right]_{kk}}{\left[F_i^{(n\times n)}\right]_{kk}\left[G_i^{(n\times n)}\right]_{kk}-\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}}\right\}\times \left\{\frac{\left[H_{i+1}^{(n\times n)}\right]_{kk}}{\left[F_{i+1}^{(n\times n)}\right]_{kk}\left[G_{i+1}^{(n\times n)}\right]_{kk}-\left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.50}$$

$$\left[Z_{38}^{(n\times n)}\right]_{kk}=-\left[J_i^{(n\times n)}\right]_{kk}\left\{\frac{\left[F_i^{(n\times n)}\right]_{kk}}{\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}-\left[F_i^{(n\times n)}\right]_{kk}\left[G_i^{(n\times n)}\right]_{kk}}\right\}\times \left\{\frac{\left[F_{i+1}^{(n\times n)}\right]_{kk}}{\left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk}-\left[G_{i+1}^{(n\times n)}\right]_{kk}\left[F_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.51}$$

$$\left[Z_{48}^{(n\times n)}\right]_{kk}=\left[J_i^{(n\times n)}\right]_{kk}\left\{\frac{\left[E_i^{(n\times n)}\right]_{kk}}{\left[E_i^{(n\times n)}\right]_{kk}\left[H_i^{(n\times n)}\right]_{kk}-\left[F_i^{(n\times n)}\right]_{kk}\left[G_i^{(n\times n)}\right]_{kk}}\right\}\times \left\{\frac{\left[F_{i+1}^{(n\times n)}\right]_{kk}}{\left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk}-\left[G_{i+1}^{(n\times n)}\right]_{kk}\left[F_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.52}$$

$$\left[Z_{77}^{(n\times n)}\right]_{kk} = -\left\{\frac{\left[H_{i+1}^{(n\times n)}\right]_{kk}}{\left[F_{i+1}^{(n\times n)}\right]_{kk}\left[G_{i+1}^{(n\times n)}\right]_{kk} - \left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.53}$$

$$\left[Z_{87}^{(n\times n)}\right]_{kk} = \left\{\frac{\left[G_{i+1}^{(n\times n)}\right]_{kk}}{\left[F_{i+1}^{(n\times n)}\right]_{kk}\left[G_{i+1}^{(n\times n)}\right]_{kk} - \left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.54}$$

$$\left[Z_{78}^{(n\times n)}\right]_{kk} = -\left\{\frac{\left[F_{i+1}^{(n\times n)}\right]_{kk}}{\left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk} - \left[G_{i+1}^{(n\times n)}\right]_{kk}\left[F_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.55}$$

$$\left[Z_{88}^{(n\times n)}\right]_{kk} = \left\{\frac{\left[E_{i+1}^{(n\times n)}\right]_{kk}}{\left[E_{i+1}^{(n\times n)}\right]_{kk}\left[H_{i+1}^{(n\times n)}\right]_{kk} - \left[G_{i+1}^{(n\times n)}\right]_{kk}\left[F_{i+1}^{(n\times n)}\right]_{kk}}\right\}, \tag{2.56}$$

where $1 \le j \le p$, $1 \le k \le n$ and, for example, $\left[Z_{88}^{(n\times n)}\right]_{kk}$ represents the kk^{th} diagonal entry of the matrix $Z_{88}^{(n\times n)}$. Thus, the streaming-plus-collision operators satisfies the condition that it be sparse and easily invertible.

While this presentation is highly abstract, the equations have a clear physical interpretation. For a two-cell problem, Eqs. (2.33) - (2.36) describe rightward streaming-plus-collision information within cell *i*; Equations (2.37) - (2.40) describe rightward streaming-plus-collision from cell *i* to cell $i+1$; Equations (2.41) - (2.44) describe leftward streaming-plus-collision information within cell *i*; Equations (2.45) - (2.48) describe rightward streaming-plus-collision information within cell $i+1$; Equations (2.49) - (2.52) describe leftward streaming-plus-collision information from cell $i+1$ to cell *i*; and, Equations (2.53) - (2.56) describe leftward streaming-plus-collision information within cell $i+1$. This type of interpretation will assist in writing the more complex two-dimensional streaming-plus-collision operator in Chapter IV.

Consider the scattering term in Eq. (2.24) for the two-cell problem defined in Eq. (2.29). Manteuffel shows that the scattering operator can be written as the product of two matrices of rank four:

$$V = \begin{bmatrix} a_i^\ell \underline{1}^{(p)} & b_i^\ell \underline{1}^{(p)} & 0 & 0 \\ c_i^\ell \underline{1}^{(p)} & d_i^\ell \underline{1}^{(p)} & 0 & 0 \\ a_i^\ell \underline{1}^{(n)} & b_i^\ell \underline{1}^{(n)} & 0 & 0 \\ c_i^\ell \underline{1}^{(n)} & d_i^\ell \underline{1}^{(n)} & 0 & 0 \\ 0 & 0 & a_{i+1}^\ell \underline{1}^{(p)} & b_{i+1}^\ell \underline{1}^{(p)} \\ 0 & 0 & c_{i+1}^\ell \underline{1}^{(p)} & d_{i+1}^\ell \underline{1}^{(p)} \\ 0 & 0 & a_{i+1}^\ell \underline{1}^{(n)} & b_{i+1}^\ell \underline{1}^{(n)} \\ 0 & 0 & c_{i+1}^\ell \underline{1}^{(n)} & d_{i+1}^\ell \underline{1}^{(n)} \end{bmatrix}, \tag{2.57}$$

$$W^T = \begin{bmatrix} \underline{w}_p^T & 0 & \underline{w}_n^T & 0 & 0 & 0 & 0 & 0 \\ 0 & \underline{w}_p^T & 0 & \underline{w}_n^T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \underline{w}_p^T & 0 & \underline{w}_n^T & 0 \\ 0 & 0 & 0 & 0 & 0 & \underline{w}_p^T & 0 & \underline{w}_n^T \end{bmatrix}. \tag{2.58}$$

The critical result of the scattering-operator decomposition of Eqs. (2.57) and (2.58) is that, regardless of quadrature order, the Sherman-Morrison decomposition will never require the inversion of a matrix, full in general, larger than four-by-four:

$$\left(I - W^T \Pi_0^{-1} V\right)^{-1} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}^{-1}, \tag{2.59}$$

where:

$$\begin{aligned} a_{11} = 1 - a_i^\ell \left[\underline{w}_p^T Z_{11}^{(p\times p)} \underline{1}^{(p)} + \underline{w}_n^T Z_{33}^{(n\times n)} \underline{1}^{(n)} \right] - \\ c_i^\ell \left[\underline{w}_p^T Z_{12}^{(p\times p)} \underline{1}^{(p)} + \underline{w}_n^T Z_{34}^{(n\times n)} \underline{1}^{(n)} \right], \end{aligned} \tag{2.60}$$

$$a_{21} = -a_i^{\ell}\left[\underline{w}_p^T Z_{21}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{43}^{(n\times n)}\underline{1}^{(n)}\right] - c_i^{\ell}\left[\underline{w}_p^T Z_{22}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{44}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.61}$$

$$a_{31} = -a_i^{\ell}\left[\underline{w}_p^T Z_{51}^{(p\times p)}\underline{1}^{(p)}\right] - c_i^{\ell}\left[\underline{w}_p^T Z_{52}^{(p\times p)}\underline{1}^{(p)}\right], \tag{2.62}$$

$$a_{41} = -a_i^{\ell}\left[\underline{w}_p^T Z_{61}^{(p\times p)}\underline{1}^{(p)}\right] - c_i^{\ell}\left[\underline{w}_p^T Z_{62}^{(p\times p)}\underline{1}^{(p)}\right], \tag{2.63}$$

$$a_{12} = -b_i^{\ell}\left[\underline{w}_p^T Z_{11}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{33}^{(n\times n)}\underline{1}^{(n)}\right] - d_i^{\ell}\left[\underline{w}_p^T Z_{12}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{34}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.64}$$

$$a_{22} = 1 - b_i^{\ell}\left[\underline{w}_p^T Z_{21}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{43}^{(n\times n)}\underline{1}^{(n)}\right] - d_i^{\ell}\left[\underline{w}_p^T Z_{22}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{44}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.65}$$

$$a_{32} = -b_i^{\ell}\left[\underline{w}_p^T Z_{51}^{(p\times p)}\underline{1}^{(p)}\right] - d_i^{\ell}\left[\underline{w}_p^T Z_{52}^{(p\times p)}\underline{1}^{(p)}\right], \tag{2.66}$$

$$a_{42} = -b_i^{\ell}\left[\underline{w}_p^T Z_{61}^{(p\times p)}\underline{1}^{(p)}\right] - d_i^{\ell}\left[\underline{w}_p^T Z_{62}^{(p\times p)}\underline{1}^{(p)}\right], \tag{2.67}$$

$$a_{13} = -a_{i+1}^{\ell}\left[\underline{w}_n^T Z_{37}^{(n\times n)}\underline{1}^{(n)}\right] - c_{i+1}^{\ell}\left[\underline{w}_n^T Z_{38}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.68}$$

$$a_{23} = -a_{i+1}^{\ell}\left[\underline{w}_n^T Z_{47}^{(n\times n)}\underline{1}^{(n)}\right] - c_{i+1}^{\ell}\left[\underline{w}_n^T Z_{48}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.69}$$

$$a_{33} = 1 - a_{i+1}^{\ell}\left[\underline{w}_p^T Z_{55}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{77}^{(n\times n)}\underline{1}^{(n)}\right] - c_{i+1}^{\ell}\left[\underline{w}_p^T Z_{56}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{78}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.70}$$

$$a_{43} = -a_{i+1}^{\ell}\left[\underline{w}_p^T Z_{65}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{87}^{(n\times n)}\underline{1}^{(n)}\right] - c_{i+1}^{\ell}\left[\underline{w}_p^T Z_{66}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{88}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.71}$$

$$a_{14} = -b_{i+1}^{\ell}\left[\underline{w}_n^T Z_{37}^{(n\times n)}\underline{1}^{(n)}\right] - d_{i+1}^{\ell}\left[\underline{w}_n^T Z_{38}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.72}$$

$$a_{24} = -b_{i+1}^{\ell}\left[\underline{w}_n^T Z_{47}^{(n\times n)}\underline{1}^{(n)}\right] - d_{i+1}^{\ell}\left[\underline{w}_n^T Z_{48}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.73}$$

$$a_{34} = -b_{i+1}^{\ell}\left[\underline{w}_p^T Z_{55}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{77}^{(n\times n)}\underline{1}^{(n)}\right] - d_{i+1}^{\ell}\left[\underline{w}_p^T Z_{56}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{78}^{(n\times n)}\underline{1}^{(n)}\right], \tag{2.74}$$

$$a_{44} = 1 - b_{i+1}^{\ell}\left[\underline{w}_p^T Z_{65}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{87}^{(n\times n)}\underline{1}^{(n)}\right] - d_{i+1}^{\ell}\left[\underline{w}_p^T Z_{66}^{(p\times p)}\underline{1}^{(p)} + \underline{w}_n^T Z_{88}^{(n\times n)}\underline{1}^{(n)}\right]. \tag{2.75}$$

Physically, Eq. (2.59) corresponds to solving for the scalar fluxes in a two-cell problem. When simplified, Eqs. (2.60) - (2.75) give the same result for the four-by-four matrix left for direct inversion reported by Manteuffel [27]. The previous discussion confirms that on the fine grid, the two-cell transport operator can be efficiently inverted. The Sherman-Morrison formula provides a detailed outline for how this inversion is completed. Now that we have constructed the two-cell operator, we will characterize the shape of the error following the two-cell inversion operating on an incident-boundary term.

Error Characterization Following Relaxation

As discussed in Chapter I, the error associated with a solution prior to convergence satisfies the homogeneous (*i.e.*, no fixed-source) transport equation. The two-cell inverse detailed in the previous section is used to relax this error. The error following relaxation will satisfy Eq. (2.76):

$$\begin{bmatrix} \underline{\psi}_i^{\ell} \\ \underline{\psi}_{i+1}^{\ell} \end{bmatrix} = \begin{bmatrix} \Pi_i^{\ell} & -\Theta_i^{\ell} \\ -\Gamma_{i+1}^{\ell} & \Pi_{i+1}^{\ell} \end{bmatrix}^{-1} \begin{bmatrix} \partial\underline{\psi}_i^{\ell} \\ \partial\underline{\psi}_{i+1}^{\ell} \end{bmatrix}. \tag{2.76}$$

The goal of this section is to describe the error following relaxation for the following types of physical problems:

- $c = 1$, and $\sigma h \gg 1$,
- $c = 1$, and $\sigma h \ll 1$,
- $c \leq 1$, and $\sigma h \gg 1$,
- $c \leq 1$, and $\sigma h \ll 1$,

- $(1-c)\sigma h \ll 1$, and $\sigma h \ll 1$,
- $(1-c)\sigma h \ll 1$, and $\sigma h \gg 1$.

As a guide for the general problem, we will assume the S_2 problem to be a sufficient estimate of the spatial variation of each angular variable in the two-cell problem. For the first two cases of problems we want to characterize, we simultaneously solve equations (2.8) - (2.11). The details will not be presented due to length. This procedure corresponds to a single relaxation step on the fine grid:

$$\begin{bmatrix} \underline{\psi}_i^{\ell} \\ \underline{\psi}_{i+1}^{\ell} \end{bmatrix}_{(1,n+1/3)} = \left[\psi_{L_i,1}^{\ell} \quad \psi_{R_i,1}^{\ell} \quad \psi_{L_i,2}^{\ell} \quad \psi_{R_i,2}^{\ell} \quad \psi_{L_{i+1},1}^{\ell} \quad \psi_{R_{i+1},1}^{\ell} \quad \psi_{L_{i+1},2}^{\ell} \quad \psi_{R_{i+1},2}^{\ell} \right]^T . \quad (2.77)$$

Manteuffel shows that for the S_2, purely scattering case $(c=1)$ and the LLD $(\theta = 1.0)$ method the error following relaxation, Eq. (2.77), is linear across each two-cell pair [27]. Our solution verifies Manteuffel's LLD result and shows that linearity holds for the family of LD equations, $1.0 \le \theta \le 3.0$. This holds for each of the two angular fluxes as well as the scalar flux.

Figure 2.3 is an example of how well error can be characterized following relaxation for purely scattering problems. This figure show that if we know edge angular flux values for a single coarse cell, we can very accurately interpolate the interface angular fluxes for a finer, two-cell problem. Figure 2.3 represents a problem that is of optically-intermediate thickness, but error-linearity holds in the thick and thin limits (*i.e.*, the first two cases in the bulleted list). This result has been demonstrated by Manteuffel [27]. Further, for optically thin problems, the error is also well characterize with a linear function across two cells [28]. When absorption is present and cells become thick, error characterization becomes more difficult. Manteuffel describes the spatial-shape of the error following relaxation as *kinked-linear* across each two-cell pair [28]. To capture this *kinked-linearity*, Manteuffel introduces a *kink-factor*.

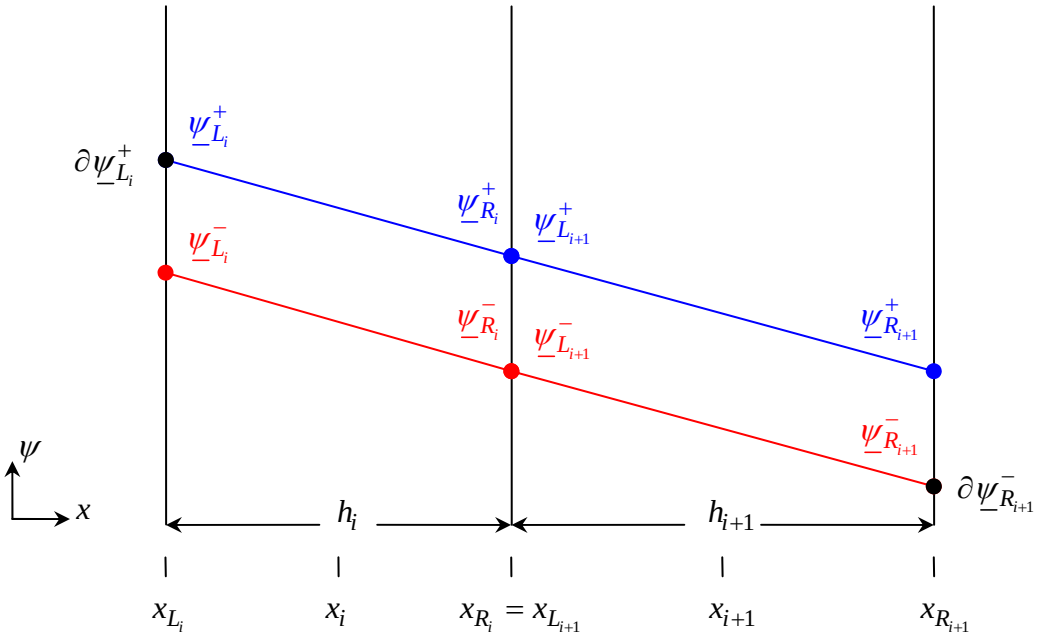

Figure 2.3 – Error Shape Following Relaxation; *c* = 1.0

Figure 2.4 is a representation of two, two-cell, S_2 problems with all properties the same except scattering ratio – one is purely-scattering $(c = 1.0)$, (*PS*) material, and one with absorption $(c < 1)$ (*WA*). A *kink-factor* attempts to measure the deviation of the with-absorption solution from simple linearity. For the homogeneous, uniform-grid problem described by Fig. 2.4, it appears that this deviation from linearity may be independent of angle – this is an assumption that Manteuffel makes and one that we adopt in this work. Given Fig. 2.4, we define our interpretation the kink-factor, Eq. (2.78), in terms of the edge-scalar-flux-contributions, Eqs. (2.79) - (2.81), to the coarse-cell-centered scalar flux:

$$\alpha = (\phi_M)\left[\frac{(\sigma_{i+1}h_{i+1}) + (\sigma_i h_i)}{(\sigma_{i+1}h_{i+1})\left[\phi_{L_i}\right]_{WA} + (\sigma_i h_i)\left[\phi_{R_{i+1}}\right]_{WA}}\right], \tag{2.78}$$

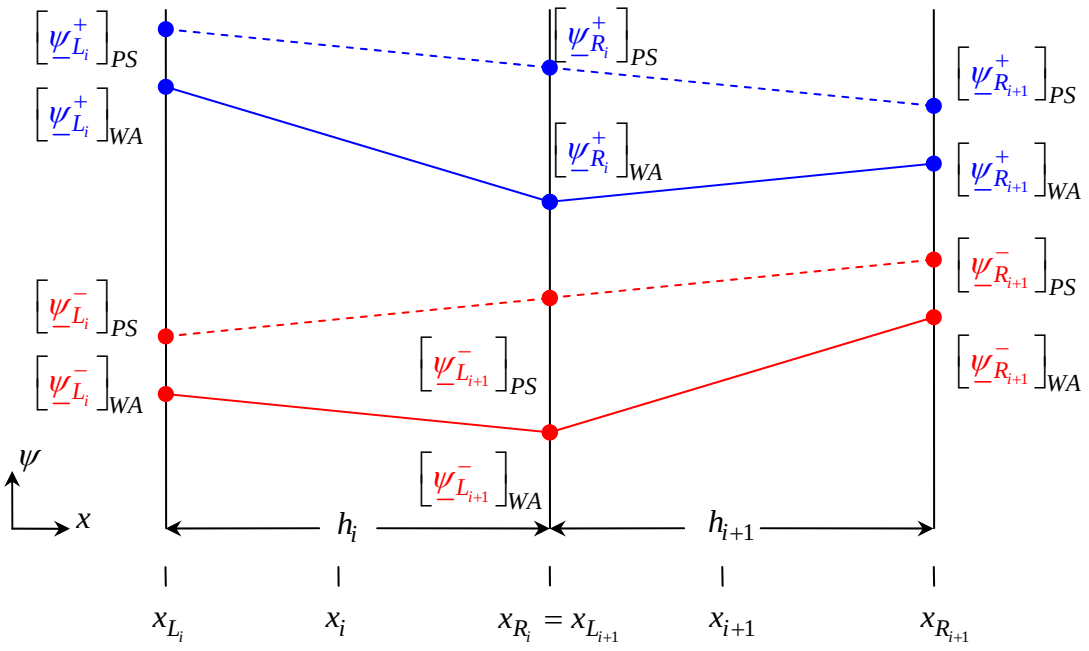

Figure 2.4 – Visualization of the *Kink-Factor*

where:

$$\phi_M = \left[\sum_{k=1}^{p} w_k \psi^{+}_{R_i,k}\right] + \left[\sum_{l=1}^{n} w_l \psi^{-}_{L_{i-1},l}\right], \tag{2.79}$$

$$\left[\phi_{L_i}\right]_{WA} = \left[\sum_{k=1}^{p} w_k \left[\psi^{+}_{L_i,k}\right]_{WA}\right] + \left[\sum_{l=1}^{n} w_l \left[\psi^{-}_{L_i,l}\right]_{WA}\right], \tag{2.80}$$

$$\left[\phi_{R_{i+1}}\right]_{WA} = \left[\sum_{k=1}^{p} w_k \left[\psi^{+}_{R_{i+1},k}\right]_{WA}\right] + \left[\sum_{l=1}^{n} w_l \left[\psi^{-}_{R_{i+1},l}\right]_{WA}\right]. \tag{2.81}$$

Figure 2.5 shows what a difficult task this characterization may be for even rather simple problems (*e.g.*, homogeneous material and uniform grid spacing). Figure 2.5 is a representation of the error following relaxation for an LLD, two-cell (each cell is 2 mean-free-paths thick), S_2 problem with a scattering ratio of 0.9. Both lines are *kinked-sub-linear*, implying a kink-factor of less that one $(\alpha < 1.0)$. Figure 2.5 also shows the

discontinuous solution will prevent our error characterization from being exact for even this simple problem. Nevertheless, we have defined a method for characterizing the error following relaxation for problems with absorption.

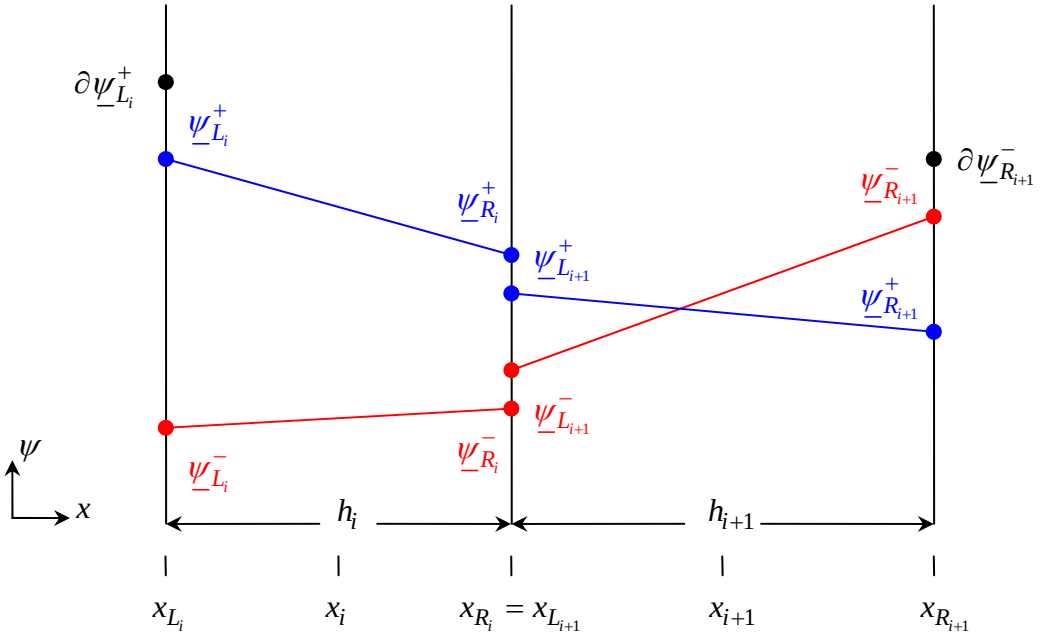

Figure 2.5 – Error Shape Following Relaxation, *c* < 1.0

Figure 2.6 shows that adjacent cells with different scattering ratios can result in error spatial variations that are angularly dependent. The solution describing the spatial variation in the angular flux residual of rightward moving particles is concave-down, $(\alpha < 1.0)$ while the analogous leftward-moving-particle-residual is concave-up $(\alpha > 1.0)$. This simple example shows how an angularly independent measure of deviation from linearity will not accurately characterize the error following relaxation. This class of problems is not considered by Manteuffel. How this class of problems affects the one-dimensional case is discussed in Chapter III.

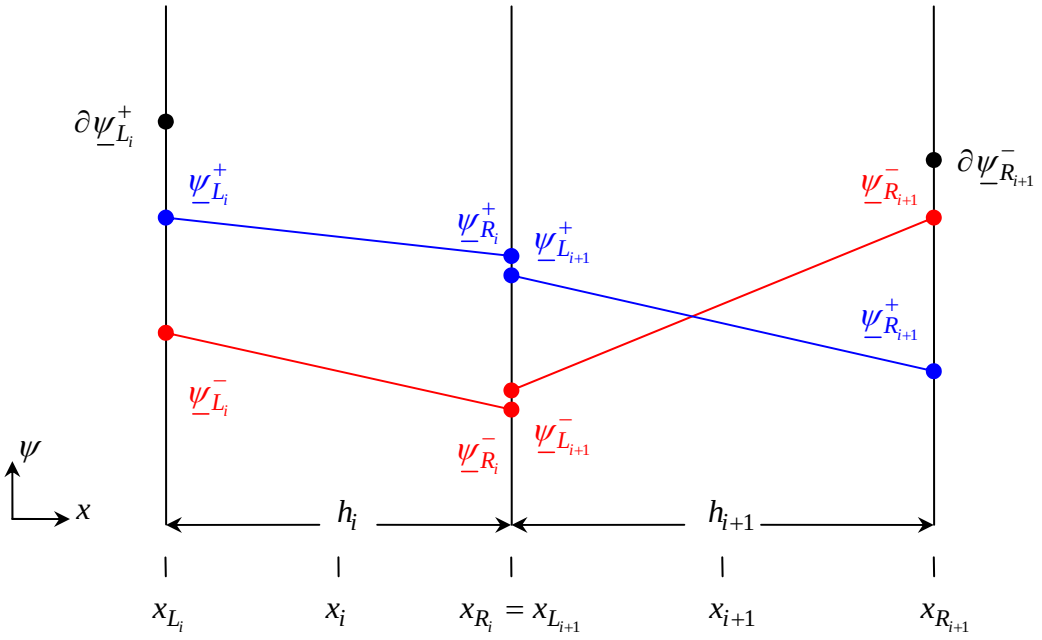

Figure 2.6 – Error Shape Following Relaxation; Heterogeneous Example

Figures 2.5 and 2.6 represent examples where Manteuffel's error characterization is invalid. Analysis of the error following relaxation in these cases does not allow for quantification of their effect on method performance. This is accomplished only through aggressive numerical experimentation. These cautionary examples notwithstanding, Manteuffel's error characterization is accurate for the first four items in our bulleted list for many of the problems he considered – recall that these first four bullets describe 1) optically thin, purely scattering; 2) optically thick, purely scattering; 3) optically thin, with absorption; and, 4) optically thick, with absorption. The final two cases we consider are of particular interest to the transport community and concentrate on the factor by which error is reduced, not the shape of that error following relaxation.

The fifth item of our bulleted list represents optically thick problems with absorption; however, the cells are thick relative to absorption mean-free paths. To describe this case, we begin by dividing the one-dimensional LDFEM equations by the

cell-thickness in mean-free paths. Following this operation, we decompose the two-cell operator of Eq. (2.29) in a particular way to obtain Eq. (2.82):

$$\begin{bmatrix} \Pi_i^{\ell} & -\Theta_i^{\ell} \\ -\Gamma_{i+1}^{\ell} & \Pi_{i+1}^{\ell} \end{bmatrix} = \mathrm{H}_0 - \mathrm{H}_1 . \tag{2.82}$$

The matrices H_0, and H_1 in Eq. (2.82) contain only scattering-rate-density and streaming-rate-density information, respectively, that has been divided by the cell-thickness in mean-free paths. Viewing Eq. (2.82) in terms of the Sherman-Morrison decomposition, Eq. (2.83), we see that the scattering-rate-density operator requires direct inversion:

$$\Pi_0 - VW^T = \mathrm{H}_0 - \mathrm{H}_1 . \tag{2.83}$$

Ultimately, we want an estimate of the error reduction associated with inverse of Eq. (2.83). It is interesting to note that this is a fundamentally different calculation from the sparse-matrix inversion detailed in the previous section. Following a great deal of algebra, we obtain Eq. (2.84) – the inverse of the collision-rate-density plus scattering-rate density operator takes the following form:

$$\mathrm{H}_0^{-1} = \begin{bmatrix} \Pi_i^{-1} & 0 \\ 0 & \Pi_{i+1}^{-1} \end{bmatrix}. \tag{2.84}$$

The diagonal blocks of Eq. (2.84) are left unspecified for the sake of brevity. The critical characteristic of Eq. (2.84) is that is it dimensionless and contains no information detailing cell thickness. This is not true of the second term in the decomposition of Eq. (2.83), H_1, which has dimensions of $(\sigma h)^{-1}$. Relative to any matrix norm $\| \cdot \|$, Eq. (2.85) is true for any physical problem:

$$\left\| \mathrm{H}_0^{-1}\mathrm{H}_1 \right\| < 1 . \tag{2.85}$$

Write the inverse of the right-hand side of Eq. (2.83) as:

$$\left(\mathrm{H}_0 - \mathrm{H}_1\right)^{-1} = \left(I - \mathrm{H}_0^{-1}\mathrm{H}_1\right)^{-1}\left(\mathrm{H}_0^{-1}\right). \tag{2.86}$$

We can write part of Eq. (2.86) in terms of a power series:

$$\left(I - \mathrm{H}_0^{-1}\mathrm{H}_1\right)^{-1} = I + \mathrm{H}_0^{-1}\mathrm{H}_1 + \left(\mathrm{H}_0^{-1}\mathrm{H}_1\right)^2 + \ldots + \left(\mathrm{H}_0^{-1}\mathrm{H}_1\right)^{n-1} + O(\sigma h)^{-n}. \tag{2.87}$$

Truncating Eq. (2.87) to a two-term approximation, we obtain Eq. (2.88):

$$\left(I - \mathrm{H}_0^{-1}\mathrm{H}_1\right)^{-1} \approx I + \mathrm{H}_0^{-1}\mathrm{H}_1 + O(\sigma h)^{-2}. \tag{2.88}$$

Since H_0^{-1} is a constant matrix, we can rewrite Eq. (2.86) as Eq. (2.89):

$$\left(I - \mathrm{H}_0^{-1}\mathrm{H}_1\right)^{-1}\left(\mathrm{H}_0^{-1}\right) = \mathrm{H}_0^{-1} + O(\sigma h)^{-1}. \tag{2.89}$$

Manteuffel shows that the L_2-norm of H_0^{-1} is bounded [28]. Thus, there exists a constant C_1, such that Eq. (2.90) is true:

$$\left(\mathrm{H}_0 - \mathrm{H}_1\right)^{-1} \le C_1. \tag{2.90}$$

Substituting Eq. (2.90) into the error equation, Eq. (2.76), and taking advantage of the properties of a matrix norm, we obtain an expression for the error reduction as a result of two-cell inversion relative to the L_2 matrix norm:

$$\left\| \begin{bmatrix} \underline{\psi}_i^{\ell} \\ \underline{\psi}_{i+1}^{\ell} \end{bmatrix} \right\|_2 \le \frac{C_2}{\sigma h} \left\| \begin{bmatrix} \partial\underline{\psi}_i^{\ell} \\ \partial\underline{\psi}_{i+1}^{\ell} \end{bmatrix} \right\|_2, \tag{2.91}$$

where the cell-thickness appearing in the denominator of Eq. (2.91)is the maximum of the two-cell pair, $\sigma h = \max\left(\sigma_i h_i, \sigma_{i+1} h_{i+1}\right)$. This error-reduction analysis is similar to that of Manteuffel [28]; however, we have extended this to include non-uniform cell-thicknesses in mean-free path space. This type of analysis indicates that the two-cell inversion step will result in a reduction of the error. This is in addition to our previous analysis which says that the shape of this reduced error can be accurately characterized for certain problems.

The sixth, and final, case in our bulleted list correspond to problems that are optically thick but are dominated by scattering. This complicates the analysis detailed in the previous discussion; however, the result is just as encouraging. For these problems, it can be shown that the error is reduced by a two-cell inverse. Again, the shape is not specified by this analysis, but the analysis does further motivate the use of a two-cell

inverse as a smoothing step. In each of the six cases in our bulleted list, we have shown that 1) the spatial shape of the error following relaxation using a two-cell inverse can be accurately characterized for certain problems, and 2) the magnitude of the error will be reduced in the thick limit for all scattering ratios. These conclusions are tempered by the examples of heterogeneous problems. As mentioned, we make no attempt to quantify the effect of heterogeneities; rather, we leave this analysis to numerical experimentation. This numerical experimentation in presented in Chapter III.

Restriction, Prolongation, and Coarse-Grid Operators

Performing the described error-smoothing step a single time results in the attenuation of error on a given grid level. To continue smoothing error at the same grid level would be inefficient since the error shape would not change significantly. To attenuate errors that are not of high frequency on the given grid level, appropriate coarse-grid operators are constructed. This section describes how these operators are built and what properties they should have.

As a multigrid procedure moves *down* the V-cycle, errors that are not attenuated at one grid level are projected onto the next coarser grid in the grid set. This projection is accomplished by a restriction operator. For the two-cell problem, the un-attenuated residual is projected onto a one-cell problem:

$$\underline{\psi}_i^{2\ell} = I_\ell^{2\ell} \begin{bmatrix} \underline{\psi}_i^{\ell} \\ \underline{\psi}_{i+1}^{\ell} \end{bmatrix} \tag{2.92}$$

The restriction operator in Eq. (2.92), $I_\ell^{2\ell}$, is constructed so that it preserves the zeroth, *i.e.*, particle conservation, and first spatial moments. This is guaranteed by constructing the restriction operator with the LD finite element weight functions. A more detailed description of the operation outlined in Eq. (2.92) is given by Eq. (2.93):

$$\begin{bmatrix} \underline{\psi}_{L_i}^{+} \\ \underline{\psi}_{R_i}^{+} \\ \underline{\psi}_{L_i}^{-} \\ \underline{\psi}_{R_i}^{-} \end{bmatrix} = \begin{bmatrix} S_1 & S_2 \end{bmatrix} \begin{bmatrix} \underline{\psi}_{L_i}^{+} & \underline{\psi}_{R_i}^{+} & \underline{\psi}_{L_i}^{-} & \underline{\psi}_{R_i}^{-} & \underline{\psi}_{L_{i+1}}^{+} & \underline{\psi}_{R_{i+1}}^{+} & \underline{\psi}_{L_{i+1}}^{-} & \underline{\psi}_{R_{i+1}}^{-} \end{bmatrix}^T . \tag{2.93}$$

The block-terms in the right-hand side of Eq. (2.93) are given by Eqs. (2.94) and (2.95):

$$S_1 = \begin{bmatrix} I^{(p\times p)} & \beta I^{(p\times p)} & 0 & 0 \\ 0 & (1-\beta) I^{(p\times p)} & 0 & 0 \\ 0 & 0 & I^{(n\times n)} & \beta I^{(n\times n)} \\ 0 & 0 & (1-\beta) I^{(n\times n)} & 0 \end{bmatrix}, \tag{2.94}$$

$$S_2 = \begin{bmatrix} \beta I^{(p\times p)} & 0 & 0 & 0 \\ (1-\beta) I^{(p\times p)} & I^{(p\times p)} & 0 & 0 \\ 0 & 0 & \beta I^{(n\times n)} & 0 \\ 0 & 0 & (1-\beta) I^{(n\times n)} & I^{(n\times n)} \end{bmatrix}. \tag{2.95}$$

Equation (2.96) describes coefficients necessary to ensure proper integration with the finite element weight functions in mean-free path space:

$$\beta = \frac{\sigma_{i+1} h_{i+1}}{\sigma_i h_i + \sigma_{i+1} h_{i+1}}, \tag{2.96}$$

Note that each row in the restriction operator, Eqs. (2.94) and (2.95), sums to two (2) – the normalized value of the quadrature weights. This preserves the zeroth spatial moment. Further, note that the columns of the restriction operator all sum to one (1). This preserves the first spatial moment.

As the multigrid procedure moves *up* the V-cycle, angular-flux-corrections are constructed from coarser-grid approximations. A prolongation operator maps an angular flux correction from one grid-level to the next finer grid-level. For the one-cell problem, angular-flux-corrections must be interpolated to describe the correction for a two-cell problem:

$$\begin{bmatrix} \underline{\psi}_i^{\ell} \\ \underline{\psi}_{i+1}^{\ell} \end{bmatrix} = I_{2\ell}^{\ell} \underline{\psi}_i^{2\ell} . \tag{2.97}$$

The accuracy of the interpolation defined in Eq. (2.97) is critical to the performance of a multigrid iterative method. As described in the previous section, this interpolation is not exact; however, a kink-factor is employed in an attempt to make it accurate. The approximate interpolation step is defined by Eq. (2.98):

$$\begin{bmatrix} \underline{\psi}_{L_i}^{+} & \underline{\psi}_{R_i}^{+} & \underline{\psi}_{L_i}^{-} & \underline{\psi}_{R_i}^{-} & \underline{\psi}_{L_{i+1}}^{+} & \underline{\psi}_{R_{i+1}}^{+} & \underline{\psi}_{L_{i+1}}^{-} & \underline{\psi}_{R_{i+1}}^{-} \end{bmatrix}^T \approx \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} \begin{bmatrix} \underline{\psi}_{L_i}^{+} \\ \underline{\psi}_{R_i}^{+} \\ \underline{\psi}_{L_i}^{-} \\ \underline{\psi}_{R_i}^{-} \end{bmatrix}, \tag{2.98}$$

where the blocks of the interpolation operator on the right-hand side of Eq. (2.98) are given by Eqs. (2.99) and (2.100):

$$T_1 = \begin{bmatrix} I^{(p\times p)} & 0 & 0 & 0 \\ \alpha\beta I^{(p\times p)} & \alpha(1-\beta) I^{(p\times p)} & 0 & 0 \\ 0 & 0 & I^{(n\times n)} & 0 \\ 0 & 0 & \alpha\beta I^{(n\times n)} & \alpha(1-\beta) I^{(n\times n)} \end{bmatrix}, \tag{2.99}$$

$$T_2 = \begin{bmatrix} \alpha\beta I^{(p\times p)} & \alpha(1-\beta) I^{(p\times p)} & 0 & 0 \\ 0 & I^{(p\times p)} & 0 & 0 \\ 0 & 0 & \alpha\beta I^{(n\times n)} & \alpha(1-\beta) I^{(n\times n)} \\ 0 & 0 & 0 & I^{(n\times n)} \end{bmatrix}. \tag{2.100}$$

The *kink*-factor that appears in Eqs. (2.99) and (2.100), α, is defined by Eq. (2.78).

As indicated, the interpolation of Eq. (2.98) is approximate; further, there are two tacit assumptions as well: 1) in each half of the discrete-ordinates space, the spatial-shape of each angular-flux correction can be described by the same kink-factor; and, 2) angular flux-corrections are continuous at the two-cell interface. The first assumption, as shown in Fig. 5, is invalid for certain problems. The second assumption is clearly invalid

since the LDFEM solution is allowed to be discontinuous. The effect of a discontinuous LLD solution is discussed by Manteuffel, and it does not unacceptably degrade the performance of his method [28]. The effect of the first assumption is neither discussed nor justified by Manteuffel. The motivation behind both assumptions is the desire for a computationally inexpensive error smoother; however, these assumptions also have interesting implications for the construction of coarse-grid operators.

As discussed by Brandt, there is no *a prior* method for constructing coarse-grid operators [7]. Manteuffel defined his coarse-grid operators in terms of the restriction and prolongation operators:

$$A^{2\ell} = I_{\ell}^{2\ell} A^{\ell} I_{2\ell}^{\ell}. \tag{2.101}$$

Justification for the method of construction described by Eq. (2.101) is implied in his proof of the following theorem [27]:

> **Theorem 4** *For the S_2 angular discretization, that is, where there is only one angle in the positive direction and one angle in the negative direction, the multigrid algorithm with two-cell red-black block μ-line relaxation will be an exact solver provided the coarsest grid is solved exactly.*

It is important to understand that Manteuffel states this proof in the context of his purely-scattering results where the interpolation of Eq. (2.98) is exact. The proof begins by examining the fine-grid error following one relaxation step described by Eq. (2.102):

$$\underline{\varphi}_{(n+1/3)}^{\ell} = \left(L^{\ell}\right)^{-1} R^{\ell} \underline{\varphi}_{(n)}^{\ell}. \tag{2.102}$$

Recall $\underline{\varphi}$ has been defined in Eq. (1.23).

$$\underline{\tilde{\varphi}}_{(n+1/3)}^{\ell} = 0 - \left(L^{\ell} - R^{\ell}\right)\left(L^{\ell}\right)^{-1} R^{\ell} \underline{\varphi}_{(n)}^{\ell}, \tag{2.103}$$

where we introduce the quantity $\underline{\tilde{\varphi}}$ to differentiate *error* from *error-residual* (Note : Eq. (2.103) is the residual calculation for the fine-grid *error*). As shown in Chapter I, the *error* satisfied the homogeneous transport equation; thus, there is no source driving the problem). This *error-residual* is now transferred to the next coarser grid through a

restriction operator and drives a coarse-grid problem. The operators in Eq. (2.102) are used to attenuate *error* on the fine-grid and are not appropriate for attenuating *error*s that vary with a frequency lower than the fine-grid spacing. To address these *errors*, we construct coarse-grid operators and use them for coarse-grid *error* relaxation described by Eq. (2.104):

$$\left(L^{2\ell} - R^{2\ell}\right)\underline{\varphi}^{2\ell}_{(n+1)} = \underline{\tilde{\varphi}}^{2\ell}_{(n)}, \tag{2.104}$$

where Eq. (2.105) defines the restricted fine-grid *error-residual*:

$$\underline{\tilde{\varphi}}^{2\ell}_{(n)} = I^{2\ell}_{\ell}\underline{\tilde{\varphi}}^{\ell}_{(n+1/3)}. \tag{2.105}$$

Solve Eq. (2.104) exactly:

$$\underline{\varphi}^{2\ell}_{(n+1)} = \left(L^{2\ell} - R^{2\ell}\right)^{-1}\underline{\tilde{\varphi}}^{2\ell}_{(n)}. \tag{2.106}$$

The quantity on the left-hand side of Eq. (2.106) is transferred to the fine-grid through a prolongation operator and is added to the *error* that remained after the first relaxation step. This procedure is summarized in Eq. (2.107):

$$\underline{\varphi}^{\ell}_{(n+2/3)} = \underline{\varphi}^{\ell}_{(n+1/3)} + I^{\ell}_{2\ell}\underline{\varphi}^{2\ell}_{(n+1)}. \tag{2.107}$$

To complete the $V(1,1)$ cycle, we perform a final relaxation on the fine grid:

$$\underline{\varphi}^{\ell}_{(n+1)} = \left(L^{\ell}\right)^{-1} R^{\ell}\left[\underline{\varphi}^{\ell}_{(n+1/3)} + I^{\ell}_{2\ell}\underline{\varphi}^{2\ell}_{(n+1)}\right]. \tag{2.108}$$

Substituting quantities defined in Eqs. (2.103) - (2.106) into Eq. (2.108), we obtain Eq. (2.109) which is an expression for the error that remains on the fine-grid following one $V(1,1)$ cycle:

$$\underline{\varphi}^{\ell}_{(n+1)} = \left(L^{\ell}\right)^{-1} R^{\ell}\left[\underline{\varphi}^{\ell}_{(n+1/3)} - I^{\ell}_{2\ell}\left(L^{2\ell} - R^{2\ell}\right)^{-1} I^{2\ell}_{\ell}\left(L^{\ell} - R^{\ell}\right)\underline{\varphi}^{\ell}_{(n+1/3)}\right]. \tag{2.109}$$

For the S_2, purely-scattering problem, the interpolation of Eq. (2.110) is exact:

$$\underline{\varphi}^{\ell}_{(n+1/3)} = I^{\ell}_{2\ell}\underline{\varphi}^{2\ell}_{(n)}, \tag{2.110}$$

Substitute this into Eq. (2.109) we obtain Eq. (2.111):

$$\underline{\varphi}^{\ell}_{(n+1)} = \left(L^{\ell}\right)^{-1} R^{\ell}\left[I^{\ell}_{2\ell} - I^{\ell}_{2\ell}\left(L^{2\ell} - R^{2\ell}\right)^{-1} I^{2\ell}_{\ell}\left(L^{\ell} - R^{\ell}\right) I^{\ell}_{2\ell} \right] \underline{\varphi}^{2\ell}_{(n)}. \tag{2.111}$$

This step reveals the importance of Manteuffel's coarse-grid operator construction defined in Eq. (2.101). Using this definition we can write the coarse-grid operator in the manner of Eq. (2.112):

$$\left(L^{2\ell} - R^{2\ell}\right) = I^{2\ell}_{\ell}\left(L^{\ell} - R^{\ell}\right) I^{\ell}_{2\ell}. \tag{2.112}$$

Substitute Eq. (2.112) into Eq. (2.111):

$$\underline{\varphi}^{\ell}_{(n+1)} = \left(L^{\ell}\right)^{-1} R^{\ell}\left[I^{\ell}_{2\ell} - I^{\ell}_{2\ell}\left(L^{2\ell} - R^{2\ell}\right)^{-1}\left(L^{2\ell} - R^{2\ell}\right) \right] \underline{\varphi}^{2\ell}_{(n)}. \tag{2.113}$$

The matrix multiplication of the coarse-grid operator and its inverse in Eq. (2.113) produces the identity matrix and the proof is complete:

$$\underline{\varphi}^{\ell}_{(n+1)} = \left(L^{\ell}\right)^{-1} R^{\ell}\left[I^{\ell}_{2\ell} - I^{\ell}_{2\ell} \right] \underline{\varphi}^{2\ell}_{(n)} = 0. \tag{2.114}$$

This proof reveals more than is stated in the corresponding theorem. First, Eq. (2.114) shows that the final relaxation step is not necessary; second, the conditions that will result in this multigrid method being an exact solver are far more restrictive than are made immediately obvious. Three elements are critical to the performance of this method: 1) the interpolation of Eq. (2.98) must be as accurate as possible; 2) coarse-grid operators must be defined in the manner of Eq. (2.101); and, 3) the coarsest grid must be solved exactly. Coarse-grid operator construction is a matter of definition. Solving the coarsest-grid problem exactly is a choice inherent in the structure of the method. Thus, the most important step in an effective multigrid method is the interpolation defined in Eq. (2.98), and this interpolation is exact *only* for S_2, purely scattering problems.

With our method for coarse-grid operator construction defined, we turn our attention to the structure of the coarse-grid operator. We want the error relaxation on coarse-grids to be as computationally efficient as the two-cell inversion described previously in Sec. II. At this point, we have no reason to expect that we can decompose coarse-grid, two-cell operators into the form required by the Sherman-Morrison formula.

Using Eq. (2.101), we can write a coarse-grid, one-cell operator from a fine-grid, two-cell operator:

$$\Pi_i^{2\ell} = I_\ell^{2\ell}\begin{bmatrix} \Pi_i^\ell & -\Theta_i^\ell \\ -\Gamma_{i+1}^\ell & \Pi_{i+1}^\ell \end{bmatrix} I_{2\ell}^\ell . \tag{2.115}$$

In terms of the Sherman-Morrison decomposition, Eq. (2.115) can be written as Eq. (2.116):

$$\Pi_i^{2\ell} = I_\ell^{2\ell}\left[\Pi_0^\ell - VW^T\right] I_{2\ell}^\ell . \tag{2.116}$$

First, we consider the construction of the coarse-grid streaming-plus-collision operator:

$$\Pi_0^{2\ell} = I_\ell^{2\ell}\Pi_0^\ell I_\ell^{2\ell} . \tag{2.117}$$

Following the matrix operations of Eq. (2.117), we obtain the following expression for a coarse, one-cell streaming-plus-collision operator, Eq. (2.118) – the individual blocks of this equations are defined in Eqs. (2.119) - (2.126):

$$\Pi_0^{2\ell} = \begin{bmatrix} A_{\hat{i}}^{(p\times p)} & B_{\hat{i}}^{(p\times p)} & 0 & 0 \\ C_{\hat{i}}^{(p\times p)} & D_{\hat{i}}^{(p\times p)} & 0 & 0 \\ 0 & 0 & E_{\hat{i}}^{(n\times n)} & F_{\hat{i}}^{(n\times n)} \\ 0 & 0 & G_{\hat{i}}^{(n\times n)} & H_{\hat{i}}^{(n\times n)} \end{bmatrix}, \tag{2.118}$$

where:

$$A_{\hat{i}}^{2\ell} = A_i^\ell + \alpha\beta B_i^\ell + \beta C_i^\ell + \alpha\beta^2 D_i^\ell + \alpha\beta^2 A_{i+1}^\ell + \alpha\beta^2 K_{i+1}^\ell . \tag{2.119}$$

$$\begin{aligned} B_{\hat{i}}^{2\ell} = {} & \alpha(1-\beta)B_i^\ell + \alpha\beta(1-\beta)D_i^\ell + \alpha\beta(1-\beta)A_{i+1}^\ell \\ & + \beta B_{i+1}^\ell + \alpha\beta(1-\beta)K_{i+1}^\ell, \end{aligned} \tag{2.120}$$

$$\begin{aligned} C_{\hat{i}}^{2\ell} = {} & (1-\beta)C_i^\ell + \alpha\beta(1-\beta)D_i^\ell + \alpha\beta(1-\beta)A_{i+1}^\ell \\ & + \alpha\beta C_{i+1}^\ell + \alpha\beta(1-\beta)K_{i+1}^\ell, \end{aligned} \tag{2.121}$$

$$\begin{aligned} D_{\hat{i}}^{2\ell} = {} & \alpha(1-\beta)^2 D_i^\ell + \alpha(1-\beta)^2 A_{i+1}^\ell + (1-\beta)B_{i+1}^\ell + \\ & \alpha(1-\beta)C_{i+1}^\ell + D_{i+1}^\ell + \alpha(1-\beta)^2 K_{i+1}^\ell, \end{aligned} \tag{2.122}$$

$$E_{\hat{i}}^{2\ell} = E_i^{\ell} + \alpha\beta F_i^{\ell} + \beta G_i^{\ell} + \alpha\beta^2 H_i^{\ell} + \alpha\beta^2 J_i^{\ell} + \alpha\beta^2 E_{i+1}^{\ell}, \tag{2.123}$$

$$\begin{aligned} F_{\hat{i}}^{2\ell} = {} & \alpha(1-\beta)F_i^{\ell} + \alpha\beta(1-\beta)H_i^{\ell} + \alpha\beta(1-\beta)J_i^{\ell} \\ & + \alpha\beta(1-\beta)E_{i+1}^{\ell} + \beta F_{i+1}^{\ell}, \end{aligned} \tag{2.124}$$

$$\begin{aligned} G_{\hat{i}}^{2\ell} = {} & (1-\beta)G_i^{\ell} + \alpha\beta(1-\beta)H_i^{\ell} + \alpha\beta(1-\beta)J_i^{\ell} \\ & + \alpha\beta(1-\beta)E_{i+1}^{\ell} + \alpha\beta G_{i+1}^{\ell}, \end{aligned} \tag{2.125}$$

$$\begin{aligned} H_{\hat{i}}^{2\ell} = {} & \alpha(1-\beta)^2 H_i^{\ell} + \alpha(1-\beta)^2 J_i^{\ell} + \alpha(1-\beta)^2 E_{i+1}^{\ell} + \\ & (1-\beta)F_{i+1}^{\ell} + \alpha(1-\beta)G_{i+1}^{\ell} + H_{i+1}^{\ell}. \end{aligned} \tag{2.126}$$

Next, consider the construction of the coarse, one-cell scattering operator:

$$\left[VW^T\right]_{\hat{i}}^{2\ell} = I_{\ell}^{2\ell}\left[VW^T\right]^{\ell} I_{\ell}^{2\ell}. \tag{2.127}$$

Following the matrix operation defined in Eq. (2.127), we obtain:

$$\left[VW^T\right]_{\hat{i}}^{2\ell} = \begin{bmatrix} a_{\hat{i}}^{2\ell}R^{(p\times p)} & b_{\hat{i}}^{2\ell}R^{(p\times p)} & a_{\hat{i}}^{2\ell}R^{(p\times n)} & b_{\hat{i}}^{2\ell}R^{(p\times n)} \\ c_{\hat{i}}^{2\ell}R^{(p\times p)} & d_{\hat{i}}^{2\ell}R^{(p\times p)} & c_{\hat{i}}^{2\ell}R^{(p\times n)} & d_{\hat{i}}^{2\ell}R^{(p\times n)} \\ a_{\hat{i}}^{2\ell}R^{(n\times p)} & b_{\hat{i}}^{2\ell}R^{(n\times p)} & a_{\hat{i}}^{2\ell}R^{(n\times n)} & b_{\hat{i}}^{2\ell}R^{(n\times n)} \\ c_{\hat{i}}^{2\ell}R^{(n\times p)} & d_{\hat{i}}^{2\ell}R^{(n\times p)} & c_{\hat{i}}^{2\ell}R^{(n\times n)} & d_{\hat{i}}^{2\ell}R^{(n\times n)} \end{bmatrix}, \tag{2.128}$$

where the individual blocks are defined by Eqs. (2.129) - (2.132):

$$a_{\hat{i}}^{2\ell} = a_i^{\ell} + (\alpha\beta)b_i^{\ell} + (\beta)c_i^{\ell} + (\alpha\beta^2)d_i^{\ell} + (\alpha\beta^2)a_{i+1}^{\ell}, \tag{2.129}$$

$$b_{\hat{i}}^{2\ell} = \left[\alpha(1-\beta)\right]b_i^{\ell} + \left[\alpha\beta(1-\beta)\right]d_i^{\ell} + \left[\alpha\beta(1-\beta)\right]a_{i+1}^{\ell} + (\beta)b_{i+1}^{\ell}, \tag{2.130}$$

$$c_{\hat{i}}^{2\ell} = (1-\beta)c_i^{\ell} + \left[\alpha\beta(1-\beta)\right]d_i^{\ell} + \left[\alpha\beta(1-\beta)\right]a_{i+1}^{\ell} + (\alpha\beta)c_{i+1}^{\ell}, \tag{2.131}$$

$$\begin{aligned} d_{\hat{i}}^{2\ell} = {} & \left[\alpha(1-\beta)^2\right]d_i^{\ell} + \left[\alpha(1-\beta)^2\right]a_{i+1}^{\ell} \\ & + (1-\beta)b_{i+1}^{\ell} + \left[\alpha(1-\beta)\right]c_{i+1}^{\ell} + d_{i+1}^{\ell}. \end{aligned} \tag{2.132}$$

The final two coarse-grid operators to be constructed are the coarse, two-cell boundary communication operators, Eqs. (2.133) and (2.135):

$$\Theta_{\hat{i}}^{2\ell} = I_{\ell}^{2\ell}\begin{bmatrix} 0 & 0 \\ \Theta_{i+1}^{\ell} & 0 \end{bmatrix} I_{2\ell}^{\ell}. \tag{2.133}$$

The result of Eq. (2.133) is given by Eq. (2.134):

$$\Theta_{\hat{i}}^{2\ell} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2\mathrm{N}^{(n\times n)} & 0 & 0 \end{bmatrix}. \tag{2.134}$$

The left boundary communication operator is given by Eq. (2.135):

$$\Gamma_{\hat{i}+1}^{2\ell} = I_{\ell}^{2\ell}\begin{bmatrix} 0 & \Gamma_{i+2}^{\ell} \\ 0 & 0 \end{bmatrix} I_{2\ell}^{\ell}. \tag{2.135}$$

The result of Eq. (2.135) is given by Eq. (2.136):

$$\Gamma_{\hat{i}}^{2\ell} = \begin{bmatrix} 0 & 2\mathrm{N}^{(p\times p)} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.136}$$

The simplicity of the coarse-grid boundary communication operators is unique. Unlike the streaming-plus-collision and scattering operators, the boundary communication operators have the same physical interpretation on all grids. This is not true of the operators that describe the within-cell transport equations. Despite this dissimilarity, all of the coarse-grid operators have the same structure as the fine-grid operators. As a result, the same methodology for two-cell inversion can be used on all grid levels.

A final comment on coarse-grid construction has to do with the possibility of multiple kink-factors. Multiple kink-factors could reflect either angular dependence or discontinuities at the two-cell interface. Without considering the effect this would have on the numerical performance of the multigrid method, a serious consequence can be seen in construction of the coarse-grid scattering operator. A prolongation with a simple notion of angular dependence can take the form of Eq. (2.137):

$$\begin{bmatrix} \underline{\psi}_{L_i}^{+} & \underline{\psi}_{R_i}^{+} & \underline{\psi}_{L_i}^{-} & \underline{\psi}_{R_i}^{-} & \underline{\psi}_{L_{i+1}}^{+} & \underline{\psi}_{R_{i+1}}^{+} & \underline{\psi}_{L_{i+1}}^{-} & \underline{\psi}_{R_{i+1}}^{-} \end{bmatrix}^T \approx \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} \begin{bmatrix} \underline{\psi}_{L_i}^{+} \\ \underline{\psi}_{R_i}^{+} \\ \underline{\psi}_{L_i}^{-} \\ \underline{\psi}_{R_i}^{-} \end{bmatrix}, \tag{2.137}$$

where the blocks of the interpolation operator with angularly-dependant kink-factors are given by Eqs. (2.138) and (2.139):

$$T_1 = \begin{bmatrix} I^{(p\times p)} & 0 & 0 & 0 \\ \alpha^{+}\beta I^{(p\times p)} & \alpha^{+}(1-\beta) I^{(p\times p)} & 0 & 0 \\ 0 & 0 & I^{(n\times n)} & 0 \\ 0 & 0 & \alpha^{-}\beta I^{(n\times n)} & \alpha^{-}(1-\beta) I^{(n\times n)} \end{bmatrix}, \tag{2.138}$$

$$T_2 = \begin{bmatrix} \alpha^{+}\beta I^{(p\times p)} & \alpha^{+}(1-\beta) I^{(p\times p)} & 0 & 0 \\ 0 & I^{(p\times p)} & 0 & 0 \\ 0 & 0 & \alpha^{-}\beta I^{(n\times n)} & \alpha^{-}(1-\beta) I^{(n\times n)} \\ 0 & 0 & 0 & I^{(n\times n)} \end{bmatrix}. \tag{2.139}$$

Following the construction of the coarse-grid scattering operator in the manner of Eq. (2.127), we find that the resulting matrix, Eq. (2.128), is no longer of rank two (2) but of rank four (4). This increased rank causes the smoothing step to become more computationally expensive. Consequently, if we try to make the kink-factors angularly-dependent, coarse-operator construction will require a more detailed look; however, we present results in Chapter III that indicate that this may not be worth the effort.

Summary of Chapter II

The family of Linear Discontinuous Finite Elements (LDFEMs) is chosen to discretize the spatial variable of the one-dimensional discrete-ordinate transport equation. The set of weight and basis functions that describe the LDFEM guarantee the properties of resolution, robustness, accuracy, and the preservation of boundary conditions. Further, we know this discretization will capture the thick-diffusive limit.

With this choice of spatial-discretization, we define an error smoothing step that involves inverting a two-cell transport operator. The Sherman-Morrison formula describes an algorithm for efficiently inverting the two-cell operator. Sherman-Morrison is merely a description of the steps involved. The two-cell inversion step can be interpreted physically as solving for the two-cell scalar fluxes, then using the scalar fluxes to define the total source, and then solving for the angular fluxes (one at a time) with a known total source. The error that drives each coarser-grid problem is the error that exists on the boundary of each two-cell problem on the finer-grid. The two-cell inverse attenuates the error. More importantly, we can characterize the error following relaxation accurately for certain types of problems. This gives us a prescription for building prolongation operators that will interpolate finer-grid, two-cell information from coarser-grid, one-cell information. Restriction operators are constructed such that they preserve the zeroth and first spatial moments of the transport equation guaranteeing particle balance on coarser-grids.

Initially, we have an unlimited choice of how to build coarse-grid operators; however, given a highly accurate interpolation methodology, a convincing argument can be made for how they should be constructed. They are built from fine-grid, restriction, and prolongation operators. If the multigrid method solves the coarsest grid exactly, and the coarse grid operators are defined as stated, then the performance of the multigrid method is determined by the accuracy of the interpolation. For one-dimensional, S_2, purely-scattering problems, this interpolation is exact. For problems with absorption, we try to capture the spatial-shape of the angular flux error with the introduction of a kink-factor. This interpolation can be quite accurate, however, we can certainly construct cases for which the assumptions in the prolongation operator are violated.

Coarse-grid operators have the same structure as the fine-grid operators; thus, the methods for operator decomposition and inversion are the same for fine and coarse-grid operators. Building these operators reveals an interesting constraint on the kink-factors. Not only does the single-kink-factor assumption result in a simple prolongation operator construction, but it also results in a simpler method of coarse-grid construction. If the

kink-factors were made angularly dependant, a great deal more thought would have to go into operator coarsening. On a simplistic level, angularly dependent kink-factors appear to change the rank of the scattering operator with each level of operator coarsening. This would violate the conditions of the Sherman-Morrison decomposition and cause the error relaxation step to become more computationally expensive. A detailed look at this aspect of the problem may result in a solution; however, it may require a great deal of effort for insufficient method improvement – evidence of this claim is given in Chapter III.

Chapter III presents numerical results of the method described in Chapter II. The structure of the code is based on the $V(\nu_1,\nu_2)$-cycle described in Chapter I (see Figure 1.1). This structure leads to our implementation of the multigrid method in a general solver. A flowchart of the general solver is given in Figure 2.7. Chapter III also describes a Fourier analysis method for producing theoretical predictions of the computational performance. Manteuffel presents convergence ratios based on a full multigrid solver. While these are valid measurements, a Fourier analysis provides a theoretical basis that will buttress the computational results. Further, it allows us to examine the error modes associated with the Fourier decomposition and determine how well our method, and our interpretation of Manteuffel's method, addresses the slowest-converging mode(s). Our numerical results are similar to those of Manteuffel for the problems he considered; however, for heterogeneous and strongly non-uniform-grid problems, our numerical results, confirmed by Fourier results, show that Manteuffel's error characterization does not generalize to such problems. While this appears to be a method failure, our Fourier analysis further shows that this multigrid method may be an excellent preconditioner for a Krylov solver.

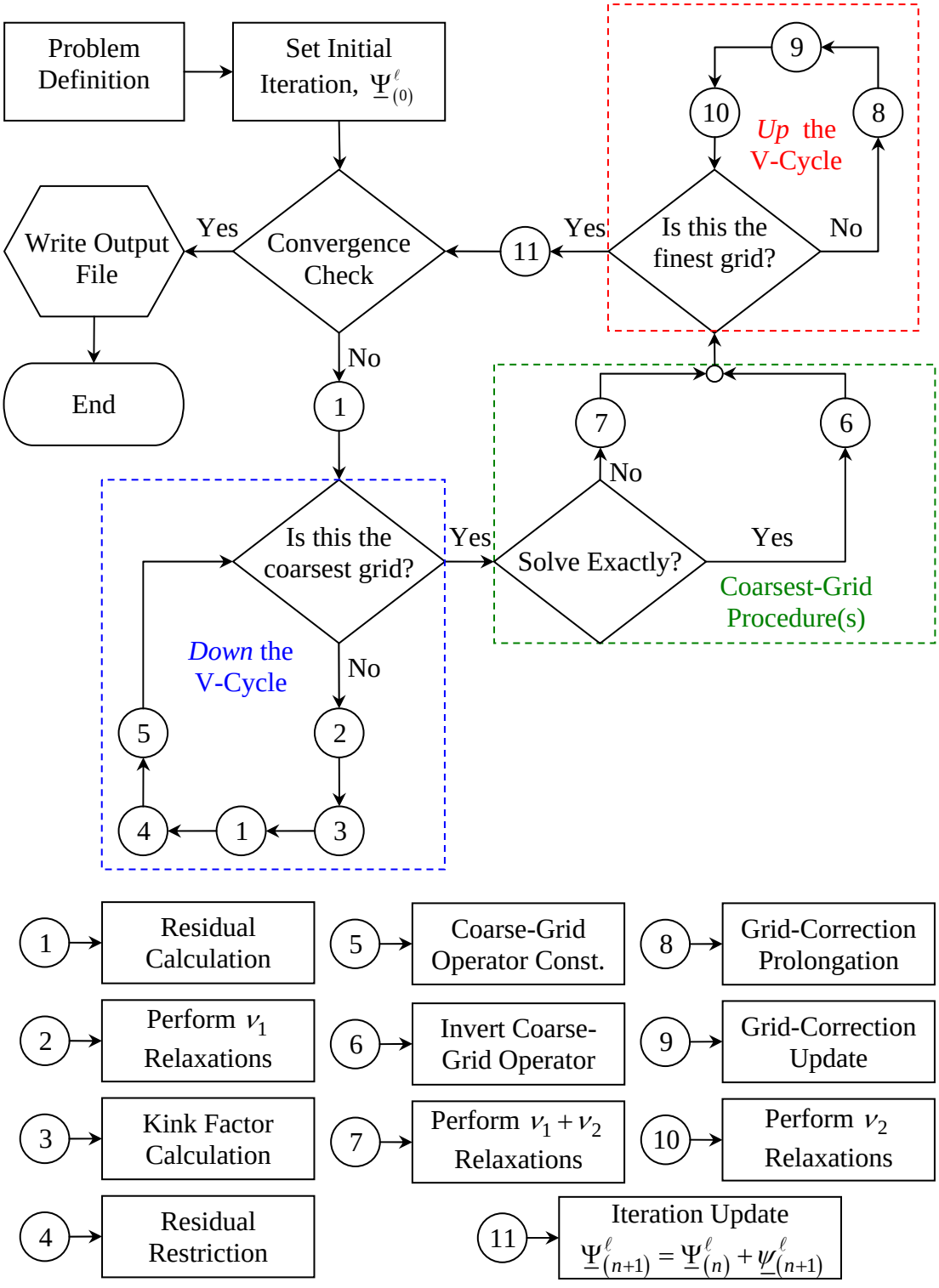

Figure 2.7 - S_N Multigrid Solver (S_NAPPER) Flowchart

CHAPTER III

ONE-DIMENSIONAL NUMERICAL RESULTS

Introduction to Chapter III

Prior to convergence, an iterative solution will contain error that can be decomposed into modes of varying frequencies. After several iterations, the remaining error mode that is least attenuated by the error smoothing step of the multigrid method will determine the convergence behavior of the iterative scheme. Manteuffel provides numerical results that estimate the spectral radius of his method. In this chapter, we present a Fourier analysis of our one-dimensional multigrid scheme. Fourier analysis predicts the convergence behavior by calculating the spectral radius of an iteration matrix that we use to validate those presented by Manteuffel and that we obtain with a general S_N multigrid solver, S_NAPPER_1D. Results of this analysis are first presented theoretically to test our understanding of the complete iterative scheme. We then present numerical results.

Theoretical Results of the One-Dimensional Fourier Analysis

Fourier analysis examines the convergence behavior of an iterative scheme that is solving a discrete problem on a source-free, infinite medium. This analysis decomposes the error into Fourier modes. Each of these modes evolve independently of any other modes. Consequently, we can examine the action of our iterative method on one error mode at a time. To examine this action we make an *ansatz* that the error following n iterations will take have the form of Eq. (3.1):

$$\underline{\Psi}^{\ell} = (\omega)^n \, \underline{a}^{\ell}, \tag{3.1}$$

where $\underline{a}^{\ell}$ is the error exiting the left and right boundaries of the eight-cell problem. This exiting information is used to construct periodic boundary conditions (*i.e.*, what exits the left boundary is used as incident information on the right boundary). We use these

periodic boundary conditions to mimic an infinite medium. This corresponds to a three-level, $V(\nu_1,\nu_2)$ multigrid cycle.

We begin the derivation of our three-level multigrid-iteration-matrix by writing an eight-cell transport problem in operator notation:

$$\left(L^{\ell} - R^{\ell}\right)\underline{\Psi}^{\ell} = \underline{Q}^{\ell}, \tag{3.2}$$

where the fixed-source vector in Eq. (3.2), $\underline{Q}^{\ell}$, contains only the boundary conditions. We expand the notation of Eq. (3.2) to show individual cell operators in Eq. (3.3):

$$\begin{bmatrix} \Pi_1^{\ell} & -\Theta_1^{\ell} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\Gamma_2^{\ell} & \Pi_2^{\ell} & -\Theta_2^{\ell} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\Gamma_3^{\ell} & \Pi_3^{\ell} & -\Theta_3^{\ell} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\Gamma_4^{\ell} & \Pi_4^{\ell} & -\Theta_4^{\ell} & 0 & 0 & 0 \\ 0 & 0 & 0 & -\Gamma_5^{\ell} & \Pi_5^{\ell} & -\Theta_5^{\ell} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\Gamma_6^{\ell} & \Pi_6^{\ell} & -\Theta_6^{\ell} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\Gamma_7^{\ell} & \Pi_7^{\ell} & -\Theta_7^{\ell} \\ 0 & 0 & 0 & 0 & 0 & 0 & -\Gamma_8^{\ell} & \Pi_8^{\ell} \end{bmatrix} \begin{bmatrix} \underline{\Psi}_1^{\ell} \\ \underline{\Psi}_2^{\ell} \\ \underline{\Psi}_3^{\ell} \\ \underline{\Psi}_4^{\ell} \\ \underline{\Psi}_5^{\ell} \\ \underline{\Psi}_6^{\ell} \\ \underline{\Psi}_7^{\ell} \\ \underline{\Psi}_8^{\ell} \end{bmatrix} = \begin{bmatrix} \underline{\Psi}_{\partial 1}^{\ell} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \underline{\Psi}_{\partial 8}^{\ell} \end{bmatrix}. \tag{3.3}$$

Equation (3.3) represents a finite eight-cell problem. To mimic an infinite medium, we incorporate periodic boundary conditions into Eq. (3.3). Periodic boundary conditions are applied by writing the right-hand side of Eq. (3.3) as boundary communication operators times the flux exiting the left and right sides of the eight-cell domain:

$$\begin{bmatrix} \underline{\Psi}^{\ell}_{\partial 1} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \underline{\Psi}^{\ell}_{\partial 8} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & F_1^{\ell} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ F_8^{\ell} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \underline{\Psi}^{\ell}_{1} \\ \underline{\Psi}^{\ell}_{2} \\ \underline{\Psi}^{\ell}_{3} \\ \underline{\Psi}^{\ell}_{4} \\ \underline{\Psi}^{\ell}_{5} \\ \underline{\Psi}^{\ell}_{6} \\ \underline{\Psi}^{\ell}_{7} \\ \underline{\Psi}^{\ell}_{8} \end{bmatrix}, \tag{3.4}$$

where the operations involving the Fourier boundary communication terms are given by Eqs. (3.5) and (3.6):

$$\underline{\Psi}^{\ell}_{\partial 1} = F_1^{\ell} \underline{\Psi}^{\ell}_{8} = \begin{bmatrix} 0 & e^{-i(\sigma\lambda)h} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \underline{\Psi}^{+}_{L_8} \\ \underline{\Psi}^{+}_{R_8} \\ \underline{\Psi}^{-}_{L_8} \\ \underline{\Psi}^{-}_{R_8} \end{bmatrix}, \tag{3.5}$$

$$\underline{\Psi}^{\ell}_{\partial 8} = F_8^{\ell} \underline{\Psi}^{\ell}_{1} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & e^{i(\sigma\lambda)h} & 0 \end{bmatrix} \begin{bmatrix} \underline{\Psi}^{+}_{L_1} \\ \underline{\Psi}^{+}_{R_1} \\ \underline{\Psi}^{-}_{L_1} \\ \underline{\Psi}^{-}_{R_1} \end{bmatrix}. \tag{3.6}$$

There are two details in Eqs. (3.5) and (3.6) that are important to understand. First, the exponents have been normalized to the width of the eight-cell problem in mean-free-path space. Second, recognize that the Fourier boundary communications operators select exiting information from the left or right boundary cell (the non-zero *column*), and incorporates that information into the boundary term for the opposite boundary (the non-zero *row*). With Eqs. (3.3) and (3.4), Eq. (3.2) can now be rewritten as Eq. (3.7):

$$\left(L^{\ell} - \tilde{R}^{\ell}\right) \underline{\Psi}^{\ell} = 0, \tag{3.7}$$

and Eq. (3.8) defines the boundary communication operator including the Fourier terms:

$$\tilde{R}^{\ell} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & F_1^{\ell} \\ 0 & 0 & \Theta_2^{\ell} & 0 & 0 & 0 & 0 & 0 \\ 0 & \Gamma_3^{\ell} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Theta_4^{\ell} & 0 & 0 & 0 \\ 0 & 0 & 0 & \Gamma_5^{\ell} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Theta_6^{\ell} & 0 \\ 0 & 0 & 0 & 0 & 0 & \Gamma_7^{\ell} & 0 & 0 \\ F_8^{\ell} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{3.8}$$

The details of the multigrid-iteration-matrix derivation will not be presented. To summarize, the solution vector in Eq. (3.7) is used as the input for the eight-cell example of Chapter I, Section V. Each eigenvalue ω, of the multigrid-iteration-matrix is computed from a single Fourier wave-number λ. Completion of one multigrid cycle produces the eigenvalue problem of Eq. (3.9):

$$A\underline{\Psi}^{\ell} = (\omega)\underline{\Psi}^{\ell}, \tag{3.9}$$

where the matrix A is defined in Eq. (3.10):

$$A = A_1 - A_2 - A_3; \tag{3.10}$$

and:

$$A_1 = \left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{(\nu_1+\nu_2)}, \tag{3.11}$$

$$A_2 = \begin{Bmatrix} \left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_2} I_{2\ell}^{\ell}\left\{\left\{\sum_{l=0}^{(\nu_1+\nu_2)}\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{l}\right\} - \left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{(\nu_1+\nu_2)}\right\}\times \\ \left(L^{2\ell}\right)^{-1} I_{\ell}^{2\ell}\left(L^{\ell} - \tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_1} \end{Bmatrix}; \tag{3.12}$$

if the coarsest grid is solved exactly:

$$A_3 = \begin{Bmatrix} \left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_2} I_{2\ell}^{\ell}\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{\nu_2} I_{4\ell}^{2\ell}\left(L^{4\ell}-\tilde{R}^{4\ell}\right)^{-1} I_{2\ell}^{4\ell}\times \\ \left\{\left(L^{2\ell}\right)\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{\nu_1}\left(L^{2\ell}\right)^{-1}\right\} I_{\ell}^{2\ell}\left(L^{\ell}-\tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_1} \end{Bmatrix}, \tag{3.13}$$

or, if further relaxations are performed on the coarsest grid:

$$A_3 = \begin{Bmatrix} \left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_2} I_{2\ell}^{\ell}\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{\nu_2} I_{4\ell}^{2\ell}\times \\ \left\{\left\{\sum_{l=0}^{(\nu_1+\nu_2)}\left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{l}\right\}-\left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{(\nu_1+\nu_2)}\right\}\times \\ I_{2\ell}^{4\ell}\left\{\left(L^{2\ell}\right)\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{\nu_1}\left(L^{2\ell}\right)^{-1}\right\} I_{\ell}^{2\ell}\left(L^{\ell}-\tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_1} \end{Bmatrix}. \tag{3.14}$$

We recognize that this presentation is highly abstract; however, a close examination of a few terms in Eqs. (3.11) – (3.14) assists in understanding the theoretical Fourier result. First, recognize the block-Jacobi-iteration defined by Eq. (3.11). For physically meaningful problems, Eq. (3.15) is always true at any grid level:

$$0 \le \left\| (L)^{-1}\left(\tilde{R}\right) \right\| < 1, \tag{3.15}$$

where $\| \cdot \|$ is a matrix norm (note that all linear algebra comments in this work are restricted to finite-dimensional vector spaces). With this in mind, consider the following term that appears in Eq. (3.12):

$$\left\{\left\{\sum_{l=0}^{(\nu_1+\nu_2)}\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{l}\right\}-\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{(\nu_1+\nu_2)}\right\}\left(L^{2\ell}\right)^{-1}. \tag{3.16}$$

Equation (3.16) is precisely, while not obviously, the matrix analog of the familiar result:

$$\lim_{N\to\infty}\left(\sum_{n=0}^{N} x^n\right) = \frac{1}{1-x};\ \text{if } 0 \le x < 1. \tag{3.17}$$

Recognizing Eq. (3.15), and applying Eq. (3.17) to (3.16) we obtain Eq. (3.18):

$$\lim_{(\nu_1+\nu_2)\to\infty}\left\{\sum_{l=0}^{(\nu_1+\nu_2)-1}\left[\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{l}\right\}=\left[I-\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{-1}. \tag{3.18}$$

Substituting the result of Eq. (3.18) into Eq. (3.16), we obtain Eq. (3.19):

$$\left[I-\left(L^{2\ell}\right)^{-1}\left(\tilde{R}^{2\ell}\right)\right]^{-1}\left(L^{2\ell}\right)^{-1}=\left(L^{2\ell}-\tilde{R}^{2\ell}\right)^{-1}. \tag{3.19}$$

The insight gained from the previous discussion reveals how the multigrid method attenuates error on the three different grid-levels. The finest grid is addressed by simple relaxation. The exact solution to intermediate grid-problems is approximated by a finite sum. The coarsest grid is either solved exactly or its solution is approximated with a finite sum.

With the derivation of the multigrid-iteration-matrix complete, we can test Manteuffel's *Theorem 4* presented in Chapter II, Section V. Testing this theorem in the context of the Fourier analysis is equivalent to setting $\nu_1=1$, and $\nu_2=0$ on the fine-grid level; and, solving grid-level 2ℓ exactly (*i.e.*, relaxing an infinite number of times):

$$A\underline{\Psi}^{\ell}=(\omega)\underline{\Psi}^{\ell}, \tag{3.20}$$

where:

$$A=A_1-A_2-A_3; \tag{3.21}$$

and:

$$A_1=\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right], \tag{3.22}$$

$$A_2=I_{2\ell}^{\ell}\left(L^{2\ell}-\tilde{R}^{2\ell}\right)^{-1}I_{\ell}^{2\ell}\left(L^{\ell}-\tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right], \tag{3.23}$$

$$A_3=\{0\}. \tag{3.24}$$

Equations (3.20) - (3.24) result in the eigenvalue problem of Eq. (3.25):

$$(\omega)\underline{\Psi}^{\ell}=\left\{\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]-I_{2\ell}^{\ell}\left(L^{2\ell}-\tilde{R}^{2\ell}\right)^{-1}I_{\ell}^{2\ell}\left(L^{\ell}-\tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]\right\}\underline{\Psi}^{\ell}. \tag{3.25}$$

With a far more general approach we arrive at Eq. (3.26), which results in the same expression as Eq. (2.111) without the unnecessary relaxation that Manteuffel's proof performs on the way *up* the V-cycle:

$$\left(\omega\right)\underline{\Psi}^{\ell}=\left\{I-I_{2\ell}^{\ell}\left(L^{2\ell}-\tilde{R}^{2\ell}\right)^{-1}I_{\ell}^{2\ell}\left(L^{\ell}-\tilde{R}^{\ell}\right)\right\}\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]\underline{\Psi}^{\ell}. \tag{3.26}$$

If the error following relaxation defined in Eq. (3.27) can be *exactly* described as interpolated error from a coarse-grid:

$$\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]\underline{\Psi}^{\ell}=I_{2\ell}^{\ell}\underline{\Psi}^{2\ell}, \tag{3.27}$$

and we define the construction of coarse-grid operators in the manner of Eq. (2.101), we obtain Eq. (3.28):

$$\left(\omega\right)\underline{\Psi}^{\ell}=\left\{I_{2\ell}^{\ell}-I_{2\ell}^{\ell}\left(L^{2\ell}-\tilde{R}^{2\ell}\right)^{-1}I_{\ell}^{2\ell}\left(L^{\ell}-\tilde{R}^{\ell}\right)I_{2\ell}^{\ell}\right\}\underline{\Psi}^{2\ell}=0. \tag{3.28}$$

If all of these conditions are satisfied, Eq. (3.28) shows the multigrid method is an exact solver. If the interpolation of Eq. (3.27) is anything other than exact, this solver will not be exact. Unlike the proof in Chapter II, Section V, the Fourier analysis can quantify (spectral radius) and describe (eigenvalues and eigenvectors) the effect of imprecise interpolation.

One-Dimensional Numerical Results – Homogeneous Material and Uniform Grid Problems

We test this multigrid method with a $V\left(1,1\right)$ cycle over a range of cell-thicknesses, quadrature orders, discretization coefficients and scattering ratios. The numerical testing begins with a homogeneous material region that is partitioned into 2^{10} (*i.e.*, 1,024 cells, or 10 grids) uniform cells. The width of the cells vary, in mean-free-path space, over nine (9) orders of magnitude, $1.0\times10^{-4}\leq\sigma_i h_i\leq1.0\times10^{4}$, in increments of an order of magnitude. Discretization parameters representing the endpoints of the range $1.0\leq\theta\leq3.0$, will be tested; however, the continuous range is a user-settable parameter. The scattering ratios tested represent problems ranging from purely-scattering

to purely-absorbing – $c \in \{1.0, 0.999999, 0.9999, 0.99, 0.9, 0.7, 0.5, 0.3, 0.1, 0.0\}$. Fourier analysis results for the S_2 case are presented in addition to angular quadrature sets of S_2, S_4, S_8, and S_{16}. The Fourier analysis yields a great deal of information; however, in the following tables, we present only the spectral radius (*i.e.*, the *predicted* convergence ratio) for this multigrid method ρ_{MG} defined by Eq. (3.29):

$$\rho_{MG} = \max_{\omega} \{\omega(\lambda)\}. \tag{3.29}$$

For the one-dimensional multigrid solver, S_NAPPER_1D, an estimate of the spectral radius (*i.e.*, the *observed* convergence ratio) is computed from the L_2 norm of the scalar flux residual at the beginning and the end of each multigrid cycle:

$$\tilde{\rho}_{MG} = \left\| \underline{\phi}_{n+1}^{\ell} \right\|_2 \Big/ \left\| \underline{\phi}_{n}^{\ell} \right\|_2 . \tag{3.30}$$

The estimate of Eq. (3.30) is obtained by specifying a problem with a solution of zero (i.e., zero incident partial current, and no fixed-source). The convergence ratio converges to a single value once all but the slowest converging error mode have been attenuated.

For the purely-scattering problems, Fourier results are not possible. There is no steady-state solution for an infinite medium problem with no loss mechanism. Consequently, the Fourier iteration matrix becomes singular and no eigenvalues are computed.

Table 3.1 - Predicted and Observed Convergence Ratios, c = 1.0, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	N/A	6.0×10^{-15}	5.8×10^{-8}	5.1×10^{-7}	2.6×10^{-6}
1.0×10^{-3}	N/A	8.2×10^{-14}	9.6×10^{-5}	3.2×10^{-4}	7.4×10^{-4}
1.0×10^{-2}	N/A	4.0×10^{-14}	1.7×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	N/A	9.4×10^{-14}	3.9×10^{-3}	4.1×10^{-3}	4.5×10^{-3}
1.0×10^{0}	N/A	3.0×10^{-11}	4.9×10^{-3}	3.8×10^{-3}	3.5×10^{-3}
1.0×10^{1}	N/A	7.5×10^{-9}	4.3×10^{-4}	3.3×10^{-4}	3.1×10^{-4}
1.0×10^{2}	N/A	1.9×10^{-6}	5.3×10^{-6}	4.2×10^{-6}	4.0×10^{-6}
1.0×10^{3}	N/A	6.4×10^{-5}	3.3×10^{-5}	3.8×10^{-5}	1.5×10^{-5}
1.0×10^{4}	N/A	4.8×10^{-3}	6.1×10^{-4}	4.2×10^{-3}	1.5×10^{-3}

Table 3.2 - Predicted and Observed Convergence Ratios, c = 1.0, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	N/A	1.1×10^{-15}	5.8×10^{-8}	5.1×10^{-7}	2.6×10^{-6}
1.0×10^{-3}	N/A	7.8×10^{-14}	9.5×10^{-5}	3.2×10^{-4}	7.4×10^{-4}
1.0×10^{-2}	N/A	3.6×10^{-14}	1.7×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	N/A	1.3×10^{-12}	3.9×10^{-3}	3.7×10^{-3}	4.5×10^{-3}
1.0×10^{0}	N/A	3.5×10^{-11}	7.1×10^{-3}	7.8×10^{-3}	6.2×10^{-3}
1.0×10^{1}	N/A	7.3×10^{-9}	1.1×10^{-3}	1.0×10^{-3}	9.7×10^{-4}
1.0×10^{2}	N/A	6.1×10^{-7}	1.1×10^{-5}	8.6×10^{-6}	8.2×10^{-6}
1.0×10^{3}	N/A	7.3×10^{-5}	3.8×10^{-5}	4.5×10^{-5}	5.3×10^{-5}
1.0×10^{4}	N/A	4.2×10^{-3}	1.0×10^{-3}	2.7×10^{-3}	2.0×10^{-3}

Table 3.3 - Predicted and Observed Convergence Ratios, c = 0.999999, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	1.8×10^{-13}	1.9×10^{-14}	5.8×10^{-8}	5.1×10^{-7}	2.7×10^{-6}
1.0×10^{-3}	3.8×10^{-13}	4.4×10^{-15}	9.6×10^{-5}	3.2×10^{-4}	7.4×10^{-4}
1.0×10^{-2}	2.7×10^{-12}	4.7×10^{-14}	1.7×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	6.9×10^{-11}	2.1×10^{-10}	3.9×10^{-3}	4.1×10^{-3}	4.6×10^{-3}
1.0×10^{0}	3.3×10^{-7}	3.9×10^{-7}	4.9×10^{-3}	3.8×10^{-3}	3.6×10^{-3}
1.0×10^{1}	9.8×10^{-6}	2.1×10^{-5}	4.3×10^{-4}	3.3×10^{-4}	3.2×10^{-4}
1.0×10^{2}	1.1×10^{-4}	1.8×10^{-4}	2.4×10^{-4}	2.3×10^{-4}	2.3×10^{-4}
1.0×10^{3}	4.1×10^{-4}	5.2×10^{-4}	5.2×10^{-4}	5.2×10^{-4}	5.1×10^{-4}
1.0×10^{4}	8.2×10^{-5}	1.1×10^{-4}	1.1×10^{-4}	1.1×10^{-4}	1.0×10^{-4}

Table 3.4 - Predicted and Observed Convergence Ratios, c = 0.999999, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	1.1×10^{-13}	6.4×10^{-15}	5.8×10^{-8}	5.1×10^{-7}	2.7×10^{-6}
1.0×10^{-3}	2.4×10^{-13}	1.9×10^{-14}	9.5×10^{-5}	3.2×10^{-4}	7.4×10^{-4}
1.0×10^{-2}	3.6×10^{-12}	4.4×10^{-14}	1.7×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	4.3×10^{-11}	7.1×10^{-11}	3.9×10^{-3}	4.0×10^{-3}	4.5×10^{-3}
1.0×10^{0}	1.9×10^{-7}	1.7×10^{-7}	7.1×10^{-3}	7.9×10^{-3}	6.2×10^{-3}
1.0×10^{1}	9.3×10^{-6}	1.4×10^{-5}	1.1×10^{-3}	1.0×10^{-3}	9.7×10^{-4}
1.0×10^{2}	1.2×10^{-4}	1.5×10^{-4}	2.1×10^{-4}	2.0×10^{-4}	2.0×10^{-4}
1.0×10^{3}	6.4×10^{-4}	6.7×10^{-4}	6.9×10^{-4}	6.8×10^{-4}	6.8×10^{-4}
1.0×10^{4}	1.6×10^{-3}	1.6×10^{-3}	1.7×10^{-3}	1.8×10^{-3}	1.8×10^{-3}

Table 3.5 - Predicted and Observed Convergence Ratios, c = 0.9999, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	8.9×10^{-14}	2.9×10^{-14}	5.8×10^{-8}	5.1×10^{-7}	2.6×10^{-6}
1.0×10^{-3}	8.7×10^{-14}	1.7×10^{-14}	9.6×10^{-5}	3.2×10^{-4}	7.4×10^{-4}
1.0×10^{-2}	1.4×10^{-10}	2.6×10^{-10}	1.7×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	1.2×10^{-7}	1.5×10^{-6}	3.9×10^{-3}	4.1×10^{-3}	4.7×10^{-3}
1.0×10^{0}	3.3×10^{-5}	1.4×10^{-4}	4.9×10^{-3}	3.8×10^{-3}	3.5×10^{-3}
1.0×10^{1}	9.4×10^{-4}	1.6×10^{-3}	2.0×10^{-3}	1.9×10^{-3}	1.9×10^{-3}
1.0×10^{2}	3.8×10^{-3}	4.6×10^{-3}	4.6×10^{-3}	4.6×10^{-3}	4.6×10^{-3}
1.0×10^{3}	5.6×10^{-4}	6.4×10^{-4}	6.1×10^{-4}	6.1×10^{-4}	6.1×10^{-4}
1.0×10^{4}	8.2×10^{-6}	8.3×10^{-6}	7.7×10^{-6}	7.6×10^{-6}	7.6×10^{-6}

Table 3.6 - Predicted and Observed Convergence Ratios, c = 0.9999, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	1.9×10^{-13}	1.3×10^{-14}	5.8×10^{-8}	5.1×10^{-7}	2.7×10^{-6}
1.0×10^{-3}	3.9×10^{-13}	1.2×10^{-14}	9.6×10^{-5}	3.2×10^{-4}	7.4×10^{-4}
1.0×10^{-2}	7.1×10^{-11}	8.8×10^{-11}	1.7×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	5.0×10^{-8}	5.3×10^{-7}	3.9×10^{-3}	3.8×10^{-3}	4.6×10^{-3}
1.0×10^{0}	1.9×10^{-5}	6.3×10^{-5}	7.1×10^{-3}	7.8×10^{-3}	6.4×10^{-3}
1.0×10^{1}	9.0×10^{-4}	1.2×10^{-3}	1.7×10^{-3}	1.6×10^{-3}	1.5×10^{-3}
1.0×10^{2}	5.9×10^{-3}	6.1×10^{-3}	6.2×10^{-3}	6.1×10^{-3}	6.1×10^{-3}
1.0×10^{3}	1.2×10^{-2}	1.2×10^{-2}	1.3×10^{-2}	1.3×10^{-2}	1.3×10^{-2}
1.0×10^{4}	2.2×10^{-2}	2.1×10^{-2}	2.1×10^{-2}	2.1×10^{-2}	2.1×10^{-2}

Table 3.7 - Predicted and Observed Convergence Ratios, c = 0.99, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	6.4×10^{-14}	1.7×10^{-14}	5.7×10^{-8}	5.6×10^{-7}	2.7×10^{-6}
1.0×10^{-3}	1.7×10^{-11}	4.0×10^{-11}	7.6×10^{-5}	3.2×10^{-4}	7.3×10^{-4}
1.0×10^{-2}	1.5×10^{-8}	1.6×10^{-6}	1.8×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	1.2×10^{-5}	3.1×10^{-4}	4.0×10^{-3}	5.0×10^{-3}	5.1×10^{-3}
1.0×10^{0}	2.8×10^{-3}	5.7×10^{-3}	5.3×10^{-3}	4.8×10^{-3}	4.6×10^{-3}
1.0×10^{1}	1.7×10^{-2}	1.8×10^{-2}	1.6×10^{-2}	1.5×10^{-2}	1.5×10^{-2}
1.0×10^{2}	7.6×10^{-4}	7.6×10^{-4}	6.9×10^{-4}	6.8×10^{-4}	6.8×10^{-4}
1.0×10^{3}	1.7×10^{-6}	1.7×10^{-6}	1.3×10^{-6}	1.2×10^{-6}	1.1×10^{-6}
1.0×10^{4}	1.9×10^{-9}	1.9×10^{-9}	1.4×10^{-9}	1.3×10^{-9}	1.2×10^{-9}

Table 3.8 - Predicted and Observed Convergence Ratios, c = 0.99, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	2.1×10^{-13}	1.9×10^{-14}	5.7×10^{-8}	5.7×10^{-7}	2.7×10^{-6}
1.0×10^{-3}	5.7×10^{-12}	1.3×10^{-11}	7.6×10^{-5}	3.2×10^{-4}	7.3×10^{-4}
1.0×10^{-2}	5.7×10^{-9}	5.4×10^{-7}	1.8×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
1.0×10^{-1}	4.9×10^{-6}	1.1×10^{-4}	4.3×10^{-3}	5.5×10^{-3}	5.7×10^{-3}
1.0×10^{0}	1.6×10^{-3}	2.8×10^{-3}	6.9×10^{-3}	7.5×10^{-3}	6.2×10^{-3}
1.0×10^{1}	2.6×10^{-2}	2.6×10^{-2}	2.4×10^{-2}	2.3×10^{-2}	2.3×10^{-2}
1.0×10^{2}	2.3×10^{-2}	2.3×10^{-2}	2.2×10^{-2}	2.1×10^{-2}	2.1×10^{-2}
1.0×10^{3}	2.1×10^{-3}	2.0×10^{-3}	1.7×10^{-3}	1.6×10^{-3}	1.6×10^{-3}
1.0×10^{4}	3.1×10^{-5}	3.1×10^{-5}	2.6×10^{-5}	2.4×10^{-5}	2.4×10^{-5}

Table 3.9 - Predicted and Observed Convergence Ratios, c = 0.9, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	3.9×10^{-13}	7.6×10^{-14}	4.3×10^{-8}	3.9×10^{-7}	2.4×10^{-6}
1.0×10^{-3}	1.5×10^{-10}	3.1×10^{-9}	5.3×10^{-5}	2.9×10^{-4}	7.0×10^{-4}
1.0×10^{-2}	1.5×10^{-7}	1.7×10^{-5}	2.1×10^{-3}	3.7×10^{-3}	4.5×10^{-3}
1.0×10^{-1}	1.1×10^{-4}	7.0×10^{-4}	5.2×10^{-3}	7.4×10^{-3}	7.8×10^{-3}
1.0×10^{0}	8.3×10^{-3}	8.9×10^{-3}	7.6×10^{-3}	6.5×10^{-3}	6.3×10^{-3}
1.0×10^{1}	4.0×10^{-3}	4.0×10^{-3}	3.1×10^{-3}	2.9×10^{-3}	2.8×10^{-3}
1.0×10^{2}	1.5×10^{-5}	1.5×10^{-5}	1.1×10^{-5}	1.0×10^{-5}	1.0×10^{-5}
1.0×10^{3}	1.7×10^{-8}	1.7×10^{-8}	1.3×10^{-8}	1.2×10^{-8}	1.2×10^{-8}
1.0×10^{4}	1.7×10^{-11}	1.7×10^{-11}	1.3×10^{-11}	1.2×10^{-11}	1.2×10^{-11}

Table 3.10 - Predicted and Observed Convergence Ratios, c = 0.9, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	4.2×10^{-12}	1.6×10^{-14}	4.3×10^{-8}	3.9×10^{-7}	2.4×10^{-6}
1.0×10^{-3}	5.8×10^{-11}	1.0×10^{-9}	5.3×10^{-5}	2.9×10^{-4}	7.0×10^{-4}
1.0×10^{-2}	5.6×10^{-8}	5.8×10^{-6}	2.1×10^{-3}	3.8×10^{-3}	4.6×10^{-3}
1.0×10^{-1}	4.3×10^{-5}	2.5×10^{-4}	5.8×10^{-3}	8.4×10^{-3}	9.2×10^{-3}
1.0×10^{0}	5.5×10^{-3}	5.8×10^{-3}	6.9×10^{-3}	7.7×10^{-3}	6.6×10^{-3}
1.0×10^{1}	1.8×10^{-2}	1.8×10^{-2}	1.7×10^{-2}	1.6×10^{-2}	1.6×10^{-2}
1.0×10^{2}	1.9×10^{-3}	1.9×10^{-3}	1.6×10^{-3}	1.6×10^{-3}	1.5×10^{-3}
1.0×10^{3}	3.0×10^{-5}	3.0×10^{-5}	2.5×10^{-5}	2.4×10^{-5}	2.3×10^{-5}
1.0×10^{4}	3.1×10^{-7}	3.1×10^{-7}	2.6×10^{-7}	2.5×10^{-7}	2.5×10^{-7}

Table 3.11 - Predicted and Observed Convergence Ratios, c = 0.7, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	1.6×10^{-12}	4.7×10^{-13}	1.9×10^{-8}	3.0×10^{-7}	1.9×10^{-6}
1.0×10^{-3}	4.6×10^{-10}	1.6×10^{-8}	3.9×10^{-5}	2.4×10^{-4}	6.1×10^{-4}
1.0×10^{-2}	4.3×10^{-7}	2.1×10^{-5}	2.2×10^{-3}	3.8×10^{-3}	4.7×10^{-3}
1.0×10^{-1}	2.4×10^{-4}	6.2×10^{-4}	3.5×10^{-3}	5.3×10^{-3}	5.7×10^{-3}
1.0×10^{0}	5.6×10^{-3}	5.7×10^{-3}	5.1×10^{-3}	4.2×10^{-3}	4.1×10^{-3}
1.0×10^{1}	6.5×10^{-4}	6.5×10^{-4}	4.8×10^{-4}	4.6×10^{-4}	4.5×10^{-4}
1.0×10^{2}	1.4×10^{-6}	1.4×10^{-6}	1.1×10^{-6}	1.0×10^{-6}	1.0×10^{-6}
1.0×10^{3}	1.5×10^{-9}	1.5×10^{-9}	1.2×10^{-9}	1.1×10^{-9}	1.1×10^{-9}
1.0×10^{4}	1.5×10^{-12}	1.5×10^{-12}	1.2×10^{-12}	1.1×10^{-12}	1.1×10^{-12}

Table 3.12 – Predicted and Observed Convergence Ratios, c = 0.7, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	5.2×10^{-13}	1.5×10^{-13}	1.9×10^{-8}	3.0×10^{-7}	1.9×10^{-6}
1.0×10^{-3}	1.7×10^{-10}	5.4×10^{-9}	3.9×10^{-5}	2.4×10^{-4}	6.1×10^{-4}
1.0×10^{-2}	1.6×10^{-7}	7.0×10^{-6}	2.3×10^{-3}	3.8×10^{-3}	4.8×10^{-3}
1.0×10^{-1}	9.7×10^{-5}	2.3×10^{-4}	4.0×10^{-3}	6.1×10^{-3}	6.9×10^{-3}
1.0×10^{0}	4.2×10^{-3}	4.2×10^{-3}	5.6×10^{-3}	5.9×10^{-3}	5.0×10^{-3}
1.0×10^{1}	6.7×10^{-3}	6.5×10^{-3}	5.8×10^{-3}	5.6×10^{-3}	5.5×10^{-3}
1.0×10^{2}	4.1×10^{-4}	4.1×10^{-4}	3.7×10^{-4}	3.6×10^{-4}	3.5×10^{-4}
1.0×10^{3}	5.1×10^{-6}	5.1×10^{-6}	4.5×10^{-6}	4.4×10^{-6}	4.4×10^{-6}
1.0×10^{4}	5.2×10^{-8}	5.2×10^{-8}	4.6×10^{-8}	4.5×10^{-8}	4.4×10^{-8}

Table 3.13 - Predicted and Observed Convergence Ratios, c = 0.5, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	1.5×10^{-12}	1.1×10^{-12}	1.5×10^{-8}	1.1×10^{-7}	1.5×10^{-6}
1.0×10^{-3}	7.6×10^{-10}	2.7×10^{-8}	3.0×10^{-5}	1.9×10^{-4}	5.3×10^{-4}
1.0×10^{-2}	7.0×10^{-7}	1.9×10^{-5}	1.9×10^{-3}	3.2×10^{-3}	4.0×10^{-3}
1.0×10^{-1}	3.0×10^{-4}	5.3×10^{-4}	3.0×10^{-3}	4.4×10^{-3}	4.9×10^{-3}
1.0×10^{0}	3.5×10^{-3}	3.5×10^{-3}	3.0×10^{-3}	2.5×10^{-3}	2.4×10^{-3}
1.0×10^{1}	2.0×10^{-4}	2.0×10^{-4}	1.5×10^{-4}	1.4×10^{-4}	1.4×10^{-4}
1.0×10^{2}	3.6×10^{-7}	3.6×10^{-7}	2.9×10^{-7}	2.9×10^{-7}	2.9×10^{-7}
1.0×10^{3}	3.8×10^{-10}	3.8×10^{-10}	3.1×10^{-10}	3.1×10^{-10}	3.1×10^{-10}
1.0×10^{4}	3.8×10^{-13}	3.8×10^{-13}	3.2×10^{-13}	3.1×10^{-13}	3.1×10^{-13}

Table 3.14 - Predicted and Observed Convergence Ratios, c = 0.5, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	1.1×10^{-12}	3.7×10^{-13}	1.5×10^{-8}	1.1×10^{-7}	1.5×10^{-6}
1.0×10^{-3}	2.9×10^{-10}	9.0×10^{-9}	3.0×10^{-5}	1.9×10^{-4}	5.2×10^{-4}
1.0×10^{-2}	2.6×10^{-7}	6.5×10^{-6}	1.9×10^{-3}	3.2×10^{-3}	4.1×10^{-3}
1.0×10^{-1}	1.2×10^{-4}	2.0×10^{-4}	3.4×10^{-3}	5.1×10^{-3}	5.9×10^{-3}
1.0×10^{0}	2.5×10^{-3}	2.5×10^{-3}	3.5×10^{-3}	3.7×10^{-3}	$3.7.\times10^{-3}$
1.0×10^{1}	4.4×10^{-3}	4.4×10^{-3}	4.0×10^{-3}	3.8×10^{-3}	$3.7.\times10^{-3}$
1.0×10^{2}	2.4×10^{-4}	2.4×10^{-4}	2.2×10^{-4}	2.2×10^{-4}	2.2×10^{-4}
1.0×10^{3}	2.9×10^{-6}	2.9×10^{-6}	2.7×10^{-6}	2.6×10^{-6}	2.6×10^{-6}
1.0×10^{4}	3.0×10^{-8}	3.0×10^{-8}	2.7×10^{-8}	2.7×10^{-8}	2.7×10^{-8}

Table 3.15 - Predicted and Observed Convergence Ratios, c = 0.3, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	2.9×10^{-12}	1.6×10^{-12}	5.6×10^{-9}	9.6×10^{-8}	9.4×10^{-7}
1.0×10^{-3}	1.1×10^{-9}	3.1×10^{-8}	2.2×10^{-5}	1.4×10^{-4}	4.3×10^{-4}
1.0×10^{-2}	9.5×10^{-7}	1.7×10^{-5}	1.6×10^{-3}	2.6×10^{-3}	3.4×10^{-3}
1.0×10^{-1}	3.2×10^{-4}	4.5×10^{-4}	2.8×10^{-3}	4.3×10^{-3}	4.7×10^{-3}
1.0×10^{0}	2.0×10^{-3}	2.0×10^{-3}	1.7×10^{-3}	1.4×10^{-3}	1.4×10^{-3}
1.0×10^{1}	6.4×10^{-5}	6.4×10^{-5}	4.9×10^{-5}	4.9×10^{-5}	5.0×10^{-5}
1.0×10^{2}	1.1×10^{-7}	1.1×10^{-7}	9.4×10^{-8}	9.6×10^{-8}	9.6×10^{-8}
1.0×10^{3}	1.2×10^{-10}	1.2×10^{-10}	1.0×10^{-10}	1.0×10^{-10}	1.0×10^{-10}
1.0×10^{4}	1.2×10^{-13}	1.2×10^{-13}	1.0×10^{-13}	1.0×10^{-13}	1.0×10^{-13}

Table 3.16 - Predicted and Observed Convergence Ratios, c = 0.3, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	8.5×10^{-13}	5.1×10^{-13}	5.6×10^{-9}	9.6×10^{-8}	9.4×10^{-7}
1.0×10^{-3}	4.0×10^{-10}	1.0×10^{-8}	2.2×10^{-5}	1.4×10^{-4}	4.2×10^{-4}
1.0×10^{-2}	3.6×10^{-7}	5.8×10^{-6}	1.6×10^{-3}	2.6×10^{-3}	3.4×10^{-3}
1.0×10^{-1}	1.3×10^{-4}	1.8×10^{-4}	3.2×10^{-3}	4.9×10^{-3}	5.7×10^{-3}
1.0×10^{0}	1.2×10^{-3}	1.2×10^{-3}	3.0×10^{-3}	3.6×10^{-3}	3.4×10^{-3}
1.0×10^{1}	3.1×10^{-3}	3.1×10^{-3}	2.9×10^{-3}	2.7×10^{-3}	2.7×10^{-3}
1.0×10^{2}	1.6×10^{-4}	1.6×10^{-4}	1.5×10^{-4}	1.5×10^{-4}	1.5×10^{-4}
1.0×10^{3}	2.0×10^{-6}	2.0×10^{-6}	1.9×10^{-6}	1.9×10^{-6}	1.9×10^{-6}
1.0×10^{4}	2.0×10^{-8}	2.0×10^{-8}	1.9×10^{-8}	1.9×10^{-8}	1.9×10^{-8}

Table 3.17 - Predicted and Observed Convergence Ratios, c = 0.1, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	2.3×10^{-12}	1.6×10^{-12}	1.9×10^{-9}	3.7×10^{-8}	4.1×10^{-7}
1.0×10^{-3}	1.4×10^{-9}	2.4×10^{-8}	1.1×10^{-5}	8.1×10^{-5}	2.9×10^{-4}
1.0×10^{-2}	1.2×10^{-6}	1.4×10^{-5}	1.2×10^{-3}	2.0×10^{-3}	2.6×10^{-3}
1.0×10^{-1}	3.2×10^{-4}	3.9×10^{-4}	2.8×10^{-3}	4.7×10^{-3}	5.0×10^{-3}
1.0×10^{0}	1.0×10^{-3}	1.0×10^{-3}	9.3×10^{-4}	1.2×10^{-3}	1.3×10^{-3}
1.0×10^{1}	1.4×10^{-5}	1.3×10^{-5}	1.1×10^{-5}	1.1×10^{-5}	1.1×10^{-5}
1.0×10^{2}	2.2×10^{-8}	2.2×10^{-8}	2.0×10^{-8}	2.1×10^{-8}	2.1×10^{-8}
1.0×10^{3}	2.4×10^{-11}	2.4×10^{-11}	2.1×10^{-11}	2.2×10^{-11}	2.3×10^{-11}
1.0×10^{4}	2.4×10^{-14}	2.4×10^{-14}	2.2×10^{-14}	2.2×10^{-14}	2.3×10^{-14}

Table 3.18 - Predicted and Observed Convergence Ratios, c = 0.1, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	9.3×10^{-13}	5.4×10^{-13}	1.9×10^{-9}	3.7×10^{-8}	4.1×10^{-7}
1.0×10^{-3}	5.1×10^{-10}	8.1×10^{-9}	1.1×10^{-5}	8.1×10^{-5}	2.9×10^{-4}
1.0×10^{-2}	4.5×10^{-7}	4.9×10^{-6}	1.2×10^{-3}	2.0×10^{-3}	2.6×10^{-3}
1.0×10^{-1}	1.4×10^{-4}	1.6×10^{-4}	3.2×10^{-3}	5.0×10^{-3}	6.2×10^{-3}
1.0×10^{0}	4.3×10^{-4}	4.3×10^{-4}	3.0×10^{-3}	4.0×10^{-3}	4.0×10^{-3}
1.0×10^{1}	5.0×10^{-4}	4.8×10^{-4}	5.2×10^{-4}	5.1×10^{-4}	5.1×10^{-4}
1.0×10^{2}	4.9×10^{-6}	4.9×10^{-6}	5.3×10^{-6}	5.3×10^{-6}	5.3×10^{-6}
1.0×10^{3}	4.6×10^{-8}	4.6×10^{-8}	4.5×10^{-8}	4.4×10^{-8}	4.4×10^{-8}
1.0×10^{4}	4.5×10^{-10}	4.5×10^{-10}	4.4×10^{-10}	4.3×10^{-10}	4.3×10^{-10}

Table 3.19 – Predicted and Observed Convergence Ratios, c = 0.0, LLD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	1.4×10^{-12}	6.7×10^{-14}	6.7×10^{-13}	7.8×10^{-12}	8.7×10^{-11}
1.0×10^{-3}	1.5×10^{-9}	1.8×10^{-10}	1.6×10^{-9}	1.5×10^{-8}	1.4×10^{-7}
1.0×10^{-2}	1.3×10^{-6}	4.7×10^{-7}	2.8×10^{-6}	2.2×10^{-5}	4.7×10^{-4}
1.0×10^{-1}	3.2×10^{-4}	3.7×10^{-4}	2.9×10^{-3}	1.0×10^{-3}	3.5×10^{-3}
1.0×10^{0}	6.5×10^{-4}	6.5×10^{-4}	7.6×10^{-4}	1.3×10^{-3}	1.4×10^{-3}
1.0×10^{1}	7.6×10^{-8}	7.5×10^{-8}	2.4×10^{-7}	2.8×10^{-7}	2.9×10^{-7}
1.0×10^{2}	1.4×10^{-13}	1.4×10^{-13}	7.0×10^{-13}	1.0×10^{-12}	1.0×10^{-12}
1.0×10^{3}	1.5×10^{-19}	2.2×10^{-16}	2.2×10^{-16}	1.4×10^{-16}	2.2×10^{-16}
1.0×10^{4}	1.5×10^{-25}	0.0	0.0	0.0	0.0

Table 3.20 - Predicted and Observed Convergence Ratios, c = 0.0, LD

$\sigma_i h_i$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	8.8×10^{-13}	2.4×10^{-14}	5.5×10^{-13}	7.0×10^{-12}	7.9×10^{-11}
1.0×10^{-3}	5.7×10^{-10}	7.6×10^{-11}	1.2×10^{-9}	1.4×10^{-8}	1.2×10^{-7}
1.0×10^{-2}	4.9×10^{-7}	2.2×10^{-7}	1.8×10^{-6}	1.8×10^{-5}	1.7×10^{-4}
1.0×10^{-1}	1.4×10^{-4}	1.5×10^{-4}	3.3×10^{-3}	1.0×10^{-3}	9.2×10^{-4}
1.0×10^{0}	1.7×10^{-4}	1.7×10^{-4}	2.9×10^{-3}	4.3×10^{-3}	4.5×10^{-3}
1.0×10^{1}	6.6×10^{-4}	6.6×10^{-4}	1.2×10^{-3}	1.4×10^{-3}	1.4×10^{-3}
1.0×10^{2}	2.2×10^{-5}	2.1×10^{-5}	6.2×10^{-5}	8.1×10^{-5}	8.7×10^{-5}
1.0×10^{3}	2.4×10^{-7}	2.4×10^{-7}	7.2×10^{-7}	9.6×10^{-7}	1.0×10^{-6}
1.0×10^{4}	2.4×10^{-9}	2.4×10^{-9}	7.3×10^{-9}	9.8×10^{-9}	1.0×10^{-8}

Tables 3.1 – 3.20 show excellent performance of this multigrid method for homogeneous problems partitioned by a uniform grid. Manteuffel examines a similar suite of problems for the LLD $(\theta = 1.0)$ discretization, and our results agree with his presentation. Our results extend Manteuffel's work to include a Fourier analysis of the S_2 problem as well as results for the LD $(\theta = 3.0)$ discretization. The application of this method to LD reveals difficulties not encountered with LLD.

The convergence ratio is a numerical estimate of the spectral radius. Both numbers allow an estimate for the number of iterations required to achieve a specified convergence criteria:

$$n = \frac{\ln(\tau)}{\ln(\rho_{MG})}, \tag{3.31}$$

where *n* is the number of iteration, τ is the convergence criteria, and ρ_{MG} is the spectral radius, or its estimate (*i.e.*, the convergence ratio). The logarithm of the convergence ratio appearing in the denominator of Eq. (3.31) shows why values approaching one (1.0), from below, are such a concern (*i.e.*, as ρ_{MG} approaches one from below, the number of iterations approaches infinity). Further, as ρ_{MG} approaches zero, from above, the number of iterations required for convergence sharply decreases.

In his paper examining purely-scattering result, Manteuffel shows the convergence ratio varies as $O\left(\sigma_i^3 h_i^3\right)$ in the thin limit, and $O\left(1/\sigma_i^2 h_i^2\right)$ in the thick limit [27]. No such deterministic relationship between convergence ratio and cell-thickness is observable in our purely scattering results (Tables 3.1 and 3.2). Our purely scattering results are obtained in the same manner as the with-absorption results. This is not true of Manteuffel's work. His purely scattering method uses cell-edge and cell-averaged quantities, and uses a different restriction operator (injection) than does his with-absorption method. Our purely scattering results reflect excellent performance in the thin limit; however, the performance degrades as the cells become optically thick. In the thick limit, the two-cell transport operator becomes ill-conditioned. This results in the

observed degraded performance – more importantly, this could produce a solution with substantial roundoff error.

With the addition of a small amount of absorption ($c = 0.999999$), our results begin to show a weak dependence on optical cell-thickness (Tables 3.3 and 3.4). In a few cases, we observe the same thin-limit performance as reported by Manteuffel. For LD, however, the performance shows that the ill-conditioning of the transport operator in the purely scattering case is observable in this weakly absorbing case. Tables 3.5 – 3.8 gradually display a clearly distinguishable correlation between optical cell-thickness and convergence ratio. Our LLD results are similar to those of Manteuffel, and they show that LD does not perform as well as LLD in the thick limit. This is not surprising since the equations describing the LLD discretization are local while those describing LD relate edge-fluxes across an entire cell.

Tables 3.7 and 3.8 contain the worst cases for both LLD and LD. Manteuffel observes his LLD worst case in the same test regime and notes that two-cell relaxation results in the most severely discontinuous interface-angular-fluxes. As a result, kinked-linear interpolation will not be as accurate. For LLD, our worst-case occurs when $c = 0.99$, and $\sigma_i h_i = 10.0$ resulting in a convergence ratio of $0.015 \le \tilde{\rho}_{MG} \le 0.018$, with a spectral radius of 0.017 determined by the Fourier analysis. For LD, our worst-case occurs when $c = 0.99$, and $\sigma_i h_i = 10.0$ resulting in a convergence ratio of $0.023 \le \tilde{\rho}_{MG} \le 0.026$, with a spectral radius of 0.026 determined by the Fourier analysis which is in agreement with the S_2 observed value.

The remaining results, Tables 3.9 – 3.20, show that for problems that become strongly absorbing, the convergence ratio varies as $O\left(\sigma_i^3 h_i^3\right)$ in the thin limit for both LD and LLD; while, in the thick limit, convergence varies as $O\left(1/\sigma_i^2 h_i^2\right)$ for LD, but varies more strongly, $O\left(1/\sigma_i^3 h_i^3\right)$, for LLD. These trends hold across the range of tested quadrature orders. These results show that we have 1) implemented Manteuffel's method correctly, 2) extended it to include a family of discontinuous finite element

discretizations, 3) verified his one-dimensional results, and, 4) most importantly, have applied a rigorous Fourier analysis that accurately predicts convergence behavior across a wide range of physical problems.

As discussed in Chapter II, Section V, coarse-grid operators are constructed from fine-grid, restriction and prolongation operators. They are not defined by the transport equation in terms of coarse-grid properties. Manteuffel implies this with the subtle statement "kink-factors are determined by the relaxation," [28]; however, the effect is not at all subtle. To demonstrate this effect, we repeat part of Manteuffel's test suite. The following results represent the convergence behavior of Manteuffel's method for slabs of thickness $\sigma_t h$, scattering ratio of $c = 0.9999$, divided into *m* cells with quadrature order S_8 (Note : Manteuffel represents the convergence ratio as 0.0 if $\rho_{MG} < 1.0 \times 10^{-11}$) [28].

We repeat the calculations of Table 3.21, however, we compute kink-factors based on coarse cell properties constructed from fine cell properties rather that coarse cell properties that result from applying standard discretizations to coarse cells. The results are summarized in Table 3.22.

The results of Table 3.22 show that our method displays better performance for problems with $4^{-5} \le \sigma_t h/2 \le 4^3$; however, our method begins to display poorer performance for $\sigma_t h/2 \ge 4^3$. The cause of this degraded performance is incorrectly computing kink-factors based on a transport discretization applied directly to the coarse cell rather than the coarse-grid operators built from fine-grid equations. This is a critical distinction as the results of Table 3.23 show.

Table 3.23 shows improved performance relative to the results of Table 3.22 for thick problems; however, our method remains slightly slower for these problems compared to Manteuffel. The difference is not unreasonable since our method uses different kink-factor calculations as well as different restriction operators.

Table 3.21 – Manteuffel's Observed Convergence Ratios, *c* = 0.9999, LLD

$\sigma_t h$	$m = 16$	$m = 64$	$m = 256$	$m = 1024$
4^{-5}	0.0	0.0	0.0	0.0
4^{-4}	3.2×10^{-11}	1.3×10^{-11}	1.0×10^{-11}	1.0×10^{-11}
4^{-3}	7.1×10^{-9}	3.0×10^{-9}	1.9×10^{-9}	2.0×10^{-9}
4^{-2}	1.2×10^{-6}	5.1×10^{-7}	3.0×10^{-7}	3.0×10^{-7}
4^{-1}	6.1×10^{-5}	2.9×10^{-5}	1.3×10^{-5}	1.1×10^{-5}
4^{0}	5.7×10^{-4}	3.7×10^{-4}	3.7×10^{-4}	3.2×10^{-4}
4^{1}	1.2×10^{-3}	1.2×10^{-3}	1.2×10^{-3}	1.1×10^{-3}
4^{2}	6.4×10^{-4}	1.2×10^{-3}	1.2×10^{-3}	1.3×10^{-3}
4^{3}	2.0×10^{-3}	1.4×10^{-3}	2.5×10^{-3}	7.1×10^{-3}
4^{4}	3.2×10^{-3}	3.8×10^{-3}	2.5×10^{-3}	3.9×10^{-3}
4^{5}	2.3×10^{-3}	3.9×10^{-3}	4.1×10^{-3}	2.6×10^{-3}
4^{6}	4.6×10^{-4}	2.3×10^{-3}	4.0×10^{-3}	4.1×10^{-3}
4^{7}	2.0×10^{-5}	4.8×10^{-4}	2.3×10^{-3}	4.0×10^{-3}
4^{8}	1.4×10^{-7}	2.1×10^{-5}	4.8×10^{-4}	2.3×10^{-3}
4^{9}	2.8×10^{-10}	1.5×10^{-7}	2.1×10^{-5}	4.8×10^{-4}
4^{10}	0.0	2.7×10^{-10}	1.5×10^{-7}	2.1×10^{-5}

Table 3.22 – S_NAPPER_1D Observed Convergence Ratios; Coarse Two-Cell Kink-Factors; c = 0.9999, LLD

$\sigma_t h$	$m = 16$	$m = 64$	$m = 256$	$m = 1024$
4^{-5}	1.3×10^{-13}	1.5×10^{-13}	1.2×10^{-13}	1.8×10^{-13}
4^{-4}	2.7×10^{-11}	7.6×10^{-11}	6.6×10^{-11}	6.5×10^{-11}
4^{-3}	4.4×10^{-9}	4.3×10^{-8}	3.7×10^{-8}	3.6×10^{-8}
4^{-2}	7.5×10^{-7}	4.2×10^{-7}	4.0×10^{-7}	4.0×10^{-7}
4^{-1}	9.1×10^{-5}	6.2×10^{-5}	6.1×10^{-5}	6.0×10^{-5}
4^{0}	2.0×10^{-3}	1.0×10^{-3}	8.8×10^{-4}	8.7×10^{-4}
4^{1}	3.6×10^{-3}	4.2×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
4^{2}	2.6×10^{-3}	4.7×10^{-3}	4.6×10^{-3}	3.3×10^{-3}
4^{3}	1.2×10^{-3}	2.9×10^{-3}	4.8×10^{-3}	4.6×10^{-3}
4^{4}	3.6×10^{-2}	3.9×10^{-2}	3.8×10^{-2}	3.8×10^{-2}
4^{5}	5.2×10^{-2}	0.12	0.12	0.12
4^{6}	6.3×10^{-3}	8.2×10^{-2}	0.16	0.16
4^{7}	3.5×10^{-4}	8.7×10^{-3}	8.9×10^{-2}	0.17
4^{8}	1.3×10^{-5}	4.1×10^{-4}	9.1×10^{-3}	9.0×10^{-2}
4^{9}	3.5×10^{-7}	1.4×10^{-5}	4.1×10^{-4}	9.2×10^{-3}
4^{10}	6.4×10^{-9}	3.6×10^{-7}	1.4×10^{-5}	4.1×10^{-4}

Table 3.23 – S_NAPPER_1D Observed Convergence Ratios; Coarse-Grid Operator Kink-Factors; c = 0.9999, LLD

$\sigma_t h$	$m = 16$	$m = 64$	$m = 256$	$m = 1024$
4^{-5}	1.4×10^{-13}	1.5×10^{-13}	1.3×10^{-13}	1.4×10^{-13}
4^{-4}	2.7×10^{-11}	7.6×10^{-11}	6.6×10^{-11}	6.6×10^{-11}
4^{-3}	4.4×10^{-9}	4.3×10^{-8}	3.7×10^{-8}	3.6×10^{-8}
4^{-2}	7.5×10^{-7}	4.2×10^{-7}	4.0×10^{-7}	4.0×10^{-7}
4^{-1}	9.1×10^{-5}	6.2×10^{-5}	6.1×10^{-5}	6.0×10^{-5}
4^{0}	2.0×10^{-3}	1.0×10^{-3}	8.8×10^{-4}	8.7×10^{-4}
4^{1}	3.5×10^{-3}	4.2×10^{-3}	2.2×10^{-3}	2.1×10^{-3}
4^{2}	2.6×10^{-3}	4.7×10^{-3}	4.6×10^{-3}	3.3×10^{-3}
4^{3}	5.2×10^{-4}	2.9×10^{-3}	4.8×10^{-3}	4.6×10^{-3}
4^{4}	2.8×10^{-3}	1.4×10^{-3}	2.9×10^{-3}	4.9×10^{-3}
4^{5}	4.0×10^{-3}	3.5×10^{-3}	1.6×10^{-3}	2.9×10^{-3}
4^{6}	1.4×10^{-3}	4.4×10^{-3}	3.5×10^{-3}	1.7×10^{-3}
4^{7}	1.9×10^{-4}	1.5×10^{-3}	4.4×10^{-3}	3.5×10^{-3}
4^{8}	1.2×10^{-5}	1.9×10^{-4}	1.5×10^{-3}	4.4×10^{-3}
4^{9}	3.4×10^{-7}	1.2×10^{-5}	1.9×10^{-4}	1.5×10^{-3}
4^{10}	6.3×10^{-9}	3.5×10^{-7}	1.2×10^{-5}	1.9×10^{-4}

One-Dimensional Numerical Results – Homogeneous Material and Non-Uniform Grid Problems

We extend our numerical experimentation to problems that represent a single material region but are partitioned with grids on non-uniform cell sizes. Particle transport is not independently sensitive to cell-thickness or cross-sections. It is the size of the cell in mean-free-paths that will affect the solution of the transport equation. As we show in the next two sections, this is distinctly different from problems that are heterogeneous.

The Fourier analysis presented in Section II of this chapter is capable of determining the convergence behavior of an infinitely repeated eight-cell pattern. We test our method in this class of problems by defining a set of eight-cell problems. We present Fourier analysis results and results from S_NAPPER_1D run with the eight-cell pattern 128 times. Figure 3.1 describes the test cases we use to test this class of problems. Table 3.24 describes the physical properties of each test problems. Tables 3.25 – 3.32 summarize the results of numerical experimentation for this problem class.

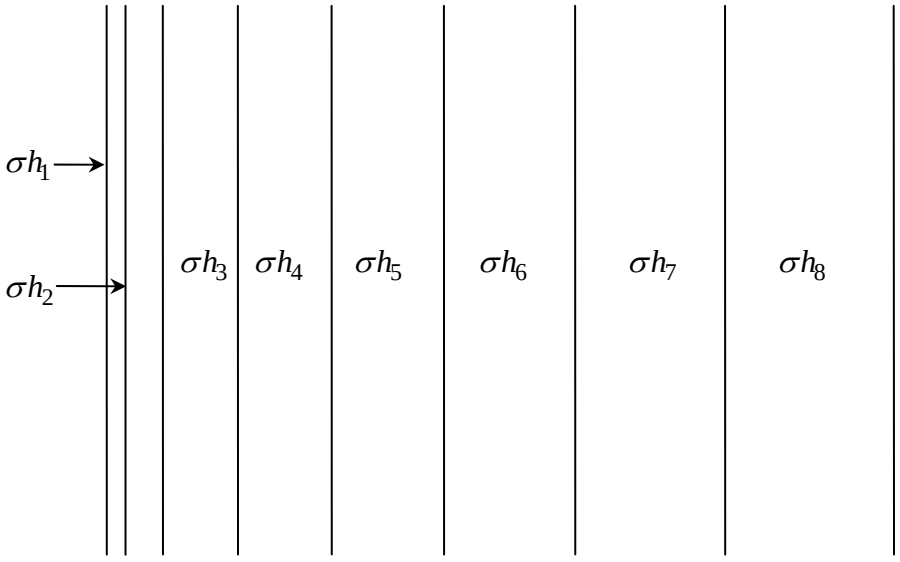

Figure 3.1 – Eight-Cell Non-Uniform Grid, Homogeneous Material Problem Set

Table 3.24 – Non-Uniform Grid Eight-Cell Arrangement

Test Problem	Cell Arrangement (Left-to-Right Ordering)							
1	σh_1	σh_2	σh_3	σh_4	σh_5	σh_6	σh_7	σh_8
2	σh_2	σh_1	σh_3	σh_4	σh_5	σh_6	σh_7	σh_8
3	σh_2	σh_3	σh_1	σh_4	σh_5	σh_6	σh_7	σh_8
4	σh_2	σh_3	σh_4	σh_1	σh_5	σh_6	σh_7	σh_8
5	σh_2	σh_3	σh_4	σh_5	σh_1	σh_6	σh_7	σh_8
6	σh_2	σh_3	σh_4	σh_5	σh_6	σh_1	σh_7	σh_8
7	σh_2	σh_3	σh_4	σh_5	σh_6	σh_7	σh_1	σh_8
8	σh_2	σh_3	σh_4	σh_5	σh_6	σh_7	σh_8	σh_1

Table 3.25 – Problem 1; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	4.8×10^{-4}	4.8×10^{-4}	2.16	2.12
0.9999	6.1×10^{-3}	6.1×10^{-3}	2.28	2.27
0.99	1.3×10^{-2}	1.3×10^{-2}	3.51	3.51
0.9	7.7×10^{-4}	7.7×10^{-4}	7.6×10^{-2}	7.6×10^{-2}
0.7	7.4×10^{-5}	7.4×10^{-5}	6.8×10^{-3}	6.8×10^{-3}
0.5	1.5×10^{-5}	1.5×10^{-5}	1.3×10^{-3}	1.3×10^{-3}
0.3	2.8×10^{-6}	2.8×10^{-6}	2.4×10^{-4}	2.4×10^{-4}
0.1	1.9×10^{-7}	1.9×10^{-7}	1.6×10^{-5}	1.6×10^{-5}
0.0	1.5×10^{-20}	0.0	9.3×10^{-11}	10^{-11}

Table 3.26 – Problem 2; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	5.5×10^{-4}	4.8×10^{-4}	2.16	2.12
0.9999	6.1×10^{-3}	6.1×10^{-3}	2.27	2.27
0.99	1.3×10^{-2}	1.3×10^{-2}	3.38	3.38
0.9	2.6×10^{-3}	2.6×10^{-3}	9.0×10^{-2}	9.0×10^{-2}
0.7	1.1×10^{-3}	1.1×10^{-3}	1.1×10^{-2}	1.1×10^{-2}
0.5	3.5×10^{-4}	3.5×10^{-4}	2.6×10^{-3}	2.6×10^{-3}
0.3	8.8×10^{-5}	8.8×10^{-5}	5.4×10^{-4}	5.4×10^{-4}
0.1	7.2×10^{-6}	7.2×10^{-6}	4.0×10^{-5}	4.0×10^{-5}
0.0	1.5×10^{-20}	0.0	9.3×10^{-11}	10^{-11}

Table 3.27 – Problem 3; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	4.6×10^{-4}	4.7×10^{-4}	2.17	2.12
0.9999	7.4×10^{-3}	7.4×10^{-3}	2.22	2.22
0.99	0.13	0.13	0.15	0.15
0.9	1.7×10^{-2}	1.7×10^{-2}	3.2×10^{-2}	3.0×10^{-2}
0.7	3.0×10^{-3}	3.0×10^{-3}	1.2×10^{-2}	1.2×10^{-2}
0.5	8.0×10^{-4}	8.0×10^{-4}	1.7×10^{-3}	1.7×10^{-3}
0.3	1.8×10^{-4}	1.8×10^{-4}	3.9×10^{-4}	3.9×10^{-4}
0.1	1.4×10^{-5}	1.4×10^{-5}	3.0×10^{-5}	3.0×10^{-5}
0.0	8.2×10^{-23}	0.0	3.8×10^{-11}	10^{-11}

Table 3.28 – Problem 4; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	4.6×10^{-4}	4.7×10^{-4}	2.16	2.12
0.9999	6.9×10^{-4}	6.9×10^{-4}	2.25	2.25
0.99	1.9×10^{-3}	1.9×10^{-3}	1.2×10^{-2}	1.2×10^{-2}
0.9	1.9×10^{-4}	1.9×10^{-4}	8.1×10^{-4}	8.1×10^{-4}
0.7	3.1×10^{-5}	3.1×10^{-5}	1.3×10^{-4}	1.3×10^{-4}
0.5	8.1×10^{-6}	8.1×10^{-6}	2.5×10^{-5}	2.5×10^{-5}
0.3	1.8×10^{-6}	1.8×10^{-6}	5.4×10^{-6}	5.4×10^{-6}
0.1	1.4×10^{-7}	1.4×10^{-7}	4.9×10^{-7}	4.9×10^{-7}
0.0	7.3×10^{-23}	0.0	3.4×10^{-11}	10^{-11}

Table 3.29 – Problem 5; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	8.0×10^{-3}	8.0×10^{-3}	2.68	2.44
0.9999	0.14	0.14	1.76	1.76
0.99	2.6×10^{-3}	2.6×10^{-3}	0.12	0.12
0.9	1.9×10^{-4}	1.9×10^{-4}	1.5×10^{-3}	1.5×10^{-3}
0.7	3.2×10^{-5}	3.2×10^{-5}	1.3×10^{-4}	1.3×10^{-4}
0.5	8.2×10^{-6}	8.2×10^{-6}	1.8×10^{-5}	1.8×10^{-5}
0.3	1.8×10^{-6}	1.8×10^{-6}	5.8×10^{-6}	10^{-6}
0.1	1.4×10^{-7}	1.4×10^{-7}	1.3×10^{-6}	10^{-6}
0.0	1.5×10^{-19}	0.0	2.8×10^{-10}	10^{-10}

Table 3.30 – Problem 6; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	6.5×10^{-4}	6.4×10^{-4}	2.57	2.42
0.9999	3.9×10^{-3}	3.9×10^{-3}	1.78	1.78
0.99	1.1×10^{-4}	1.1×10^{-4}	0.12	0.12
0.9	1.9×10^{-6}	1.9×10^{-6}	1.7×10^{-3}	1.7×10^{-3}
0.7	3.2×10^{-7}	3.2×10^{-7}	1.3×10^{-4}	1.3×10^{-4}
0.5	8.2×10^{-8}	8.2×10^{-8}	1.8×10^{-5}	1.8×10^{-5}
0.3	1.8×10^{-8}	1.8×10^{-8}	3.1×10^{-6}	3.1×10^{-6}
0.1	1.4×10^{-9}	1.4×10^{-9}	2.0×10^{-7}	2.0×10^{-7}
0.0	1.5×10^{-19}	0.0	2.8×10^{-10}	10^{-10}

Table 3.31 – Problem 7; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	0.14	0.14	1.05	1.05
0.9999	3.9×10^{-3}	3.9×10^{-3}	1.81	1.81
0.99	1.1×10^{-4}	1.1×10^{-4}	0.12	0.12
0.9	1.9×10^{-6}	1.9×10^{-6}	1.7×10^{-3}	1.7×10^{-3}
0.7	3.2×10^{-7}	3.2×10^{-7}	1.3×10^{-4}	1.3×10^{-4}
0.5	8.2×10^{-8}	8.2×10^{-8}	1.8×10^{-5}	1.8×10^{-5}
0.3	1.8×10^{-8}	1.8×10^{-8}	3.1×10^{-6}	3.1×10^{-6}
0.1	1.4×10^{-9}	1.4×10^{-9}	2.0×10^{-7}	2.0×10^{-7}
0.0	1.2×10^{-22}	0.0	1.4×10^{-11}	10^{-11}

Table 3.32 – Problem 8; Predicted and Observed Convergence Ratios

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	0.40	0.40	1.90	1.89
0.9999	0.45	0.45	1.78	1.78
0.99	0.35	0.35	0.36	0.36
0.9	0.11	0.11	0.28	0.28
0.7	2.5×10^{-2}	2.5×10^{-2}	0.30	0.30
0.5	6.9×10^{-3}	6.9×10^{-3}	3.0×10^{-2}	3.0×10^{-2}
0.3	1.6×10^{-3}	1.6×10^{-3}	5.4×10^{-3}	5.4×10^{-3}
0.1	1.2×10^{-4}	1.2×10^{-4}	3.7×10^{-4}	3.7×10^{-4}
0.0	1.2×10^{-22}	0.0	1.4×10^{-11}	10^{-11}

Tables 3.25 – 3.32 show that the Fourier analysis continues to be an accurate predictive tool for convergence behavior. Further, we immediately notice that in every example, this method diverges for highly scattering problems when LD is used as the spatial discretization. Based on the discussions of Chapter II, we have reason to examine the spatial shape of the iteration error following relaxation for these problems. The following figure (Figure 3.2) represents the error following relaxation for the rightmost two-cells in the eight cell pattern for problems 1-7.

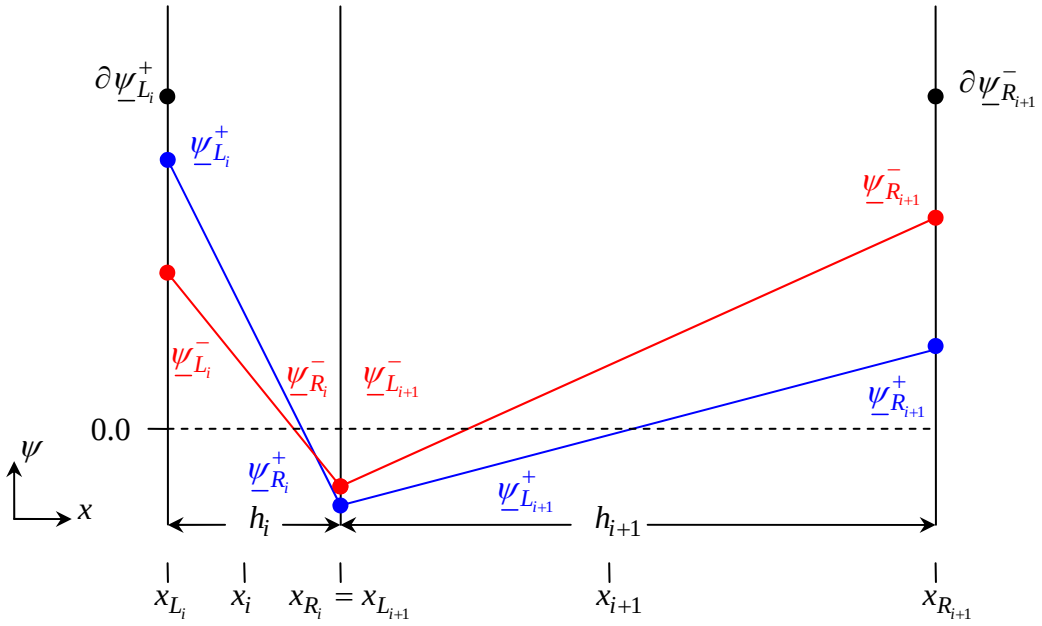

Figure 3.2 – Error Following Relaxation; Cells 7&8, Problems 1-7

Figure 3.2 shows that the two main assumptions in the prolongation operators, continuity and single kink-factor sufficiency, are not grossly violated by these problems; however, Figure 3.2 also shows that the kink-factors computed for these problems is negative. These negative kink-factors are not only involved in coarse-to-fine grid interpolation, but they are also used to build coarse-grid operators. It is reasonable to conjecture that the divergent behavior may be caused by these negative values. To test this effect, we implement a lower limit of zero for the kink-factors and rerun all the problems of Table 3.24.

Table 3.33 – Problem 1; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	5.4×10^{-4}	5.3×10^{-4}	0.33	0.32
0.9999	6.1×10^{-3}	6.1×10^{-3}	0.18	0.18
0.99	1.3×10^{-2}	1.3×10^{-2}	0.14	0.14
0.9	7.7×10^{-4}	7.7×10^{-4}	1.6×10^{-2}	1.6×10^{-2}
0.7	7.4×10^{-5}	7.4×10^{-5}	1.9×10^{-3}	1.9×10^{-3}
0.5	1.5×10^{-5}	1.5×10^{-5}	4.2×10^{-4}	4.2×10^{-4}
0.3	2.8×10^{-6}	2.8×10^{-6}	8.2×10^{-5}	8.2×10^{-5}
0.1	1.9×10^{-7}	1.9×10^{-7}	5.8×10^{-6}	5.8×10^{-6}
0.0	1.5×10^{-20}	0.0	2.1×10^{-10}	10^{-10}

Table 3.34 – Problem 2; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	6.1×10^{-4}	5.3×10^{-4}	0.33	0.32
0.9999	6.1×10^{-3}	6.1×10^{-3}	0.18	0.18
0.99	1.3×10^{-2}	1.3×10^{-2}	0.14	0.14
0.9	2.6×10^{-3}	2.6×10^{-3}	1.9×10^{-2}	1.9×10^{-2}
0.7	1.1×10^{-3}	1.1×10^{-3}	3.2×10^{-3}	3.2×10^{-3}
0.5	3.5×10^{-4}	3.5×10^{-4}	8.3×10^{-4}	8.3×10^{-4}
0.3	8.8×10^{-5}	8.8×10^{-5}	1.9×10^{-4}	1.9×10^{-4}
0.1	7.2×10^{-6}	7.2×10^{-6}	1.5×10^{-5}	1.5×10^{-5}
0.0	1.5×10^{-20}	0.0	2.1×10^{-10}	10^{-10}

Table 3.35 – Problem 3; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	5.2×10^{-4}	5.2×10^{-4}	0.32	0.32
0.9999	7.4×10^{-3}	7.4×10^{-3}	0.18	0.18
0.99	0.13	0.13	0.15	0.15
0.9	1.7×10^{-2}	1.7×10^{-2}	3.2×10^{-2}	3.2×10^{-2}
0.7	3.0×10^{-3}	3.0×10^{-3}	6.2×10^{-3}	6.2×10^{-3}
0.5	8.0×10^{-4}	8.0×10^{-4}	1.7×10^{-3}	1.7×10^{-3}
0.3	1.8×10^{-4}	1.8×10^{-4}	3.9×10^{-4}	3.9×10^{-4}
0.1	1.4×10^{-5}	1.4×10^{-5}	3.0×10^{-5}	3.0×10^{-5}
0.0	8.2×10^{-23}	0.0	5.8×10^{-11}	10^{-11}

Table 3.36 – Problem 4; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	5.2×10^{-4}	5.2×10^{-4}	0.32	0.32
0.9999	7.1×10^{-4}	7.1×10^{-4}	0.18	0.18
0.99	1.9×10^{-3}	1.9×10^{-3}	7.3×10^{-3}	7.3×10^{-3}
0.9	1.9×10^{-4}	1.9×10^{-4}	3.8×10^{-4}	3.8×10^{-4}
0.7	3.1×10^{-5}	3.1×10^{-5}	6.6×10^{-5}	6.6×10^{-5}
0.5	8.1×10^{-6}	8.1×10^{-6}	1.8×10^{-5}	1.8×10^{-5}
0.3	1.8×10^{-6}	1.8×10^{-6}	4.3×10^{-6}	4.3×10^{-6}
0.1	1.4×10^{-7}	1.4×10^{-7}	4.9×10^{-7}	4.9×10^{-7}
0.0	7.3×10^{-23}	0.0	2.4×10^{-11}	10^{-11}

Table 3.37 – Problem 5; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	8.0×10^{-3}	8.0×10^{-3}	0.28	0.29
0.9999	0.14	0.14	0.29	0.29
0.99	2.6×10^{-3}	2.6×10^{-3}	2.3×10^{-2}	2.3×10^{-2}
0.9	1.9×10^{-4}	1.9×10^{-4}	4.5×10^{-4}	4.5×10^{-4}
0.7	3.2×10^{-5}	3.2×10^{-5}	6.7×10^{-5}	6.7×10^{-5}
0.5	8.2×10^{-6}	8.2×10^{-6}	1.8×10^{-5}	1.8×10^{-5}
0.3	1.8×10^{-6}	1.8×10^{-6}	4.1×10^{-6}	4.1×10^{-6}
0.1	1.4×10^{-7}	1.4×10^{-7}	3.9×10^{-7}	4.0×10^{-7}
0.0	1.5×10^{-19}	0.0	6.5×10^{-10}	10^{-10}

Table 3.38 – Problem 6; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	6.5×10^{-4}	6.4×10^{-4}	0.31	0.31
0.9999	3.9×10^{-3}	3.9×10^{-3}	0.26	0.26
0.99	1.1×10^{-4}	1.1×10^{-4}	2.3×10^{-2}	2.3×10^{-2}
0.9	1.9×10^{-6}	1.9×10^{-6}	4.4×10^{-4}	4.4×10^{-4}
0.7	3.2×10^{-7}	3.2×10^{-7}	3.2×10^{-5}	3.2×10^{-5}
0.5	8.2×10^{-8}	8.2×10^{-8}	6.1×10^{-6}	6.1×10^{-6}
0.3	1.8×10^{-8}	1.8×10^{-8}	1.1×10^{-6}	1.1×10^{-6}
0.1	1.4×10^{-9}	1.4×10^{-9}	7.6×10^{-8}	7.5×10^{-8}
0.0	1.5×10^{-19}	0.0	6.5×10^{-10}	10^{-10}

Table 3.39 – Problem 7; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	0.14	0.14	0.34	0.34
0.9999	3.9×10^{-3}	3.9×10^{-3}	0.26	0.26
0.99	1.1×10^{-4}	1.1×10^{-4}	2.3×10^{-2}	2.3×10^{-2}
0.9	1.9×10^{-6}	1.9×10^{-6}	4.4×10^{-4}	4.4×10^{-4}
0.7	3.2×10^{-7}	3.2×10^{-7}	3.2×10^{-5}	3.2×10^{-5}
0.5	8.2×10^{-8}	8.2×10^{-8}	6.1×10^{-6}	6.1×10^{-6}
0.3	1.8×10^{-8}	1.8×10^{-8}	1.1×10^{-6}	1.1×10^{-6}
0.1	1.4×10^{-9}	1.4×10^{-9}	7.6×10^{-8}	7.5×10^{-8}
0.0	1.2×10^{-22}	0.0	2.4×10^{-11}	10^{-11}

Table 3.40 – Problem 8; Predicted and Observed Convergence Ratios; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	0.40	0.40	0.36	0.36
0.9999	0.45	0.45	0.45	0.45
0.99	0.35	0.35	0.36	0.36
0.9	0.11	0.11	0.15	0.15
0.7	2.5×10^{-2}	2.5×10^{-2}	4.1×10^{-2}	4.1×10^{-2}
0.5	6.9×10^{-3}	6.9×10^{-3}	1.2×10^{-2}	1.2×10^{-2}
0.3	1.6×10^{-3}	1.6×10^{-3}	3.0×10^{-3}	3.0×10^{-3}
0.1	1.2×10^{-4}	1.2×10^{-4}	2.4×10^{-4}	2.4×10^{-4}
0.0	1.2×10^{-22}	0.0	2.4×10^{-11}	10^{-11}

Tables 3.33 – 3.40 show that restricting the kink-factors to non-negative values does eliminate all divergent behavior observed when using the LD discretization; however, these tables also show slight changes in the convergence behavior of some problems run with LLD. Problems 1-4 show a slight increase in the convergence ratio. This change is small; however we must recognize that when we impose a restriction on the kink factor, we are restricting the method's ability to interpolate values accurately.

One-Dimensional Numerical Results – Heterogeneous Materials and Uniform Grid Problems

The next class of problem to address is a uniform grid with heterogeneous materials. All of the problems yet presented do not vary in scattering ratio. Manteuffel implies that his method is suitable for heterogeneous problems, however, he varies only cell-thickness and not scattering properties, which means he really tests only homogeneous problems with non-uniform grids [28]. We test our method with another eight-cell repeated pattern in which the scattering ratio of a single cell is gradually decreased (Figure 3.3).

For the sake of brevity, we present a set of problems that represent a serious challenge to this multigrid method rather than showing an exhaustive analysis of this large class of problems. We present the results for nine problems with the following specifications: S_2, $h = 100.0$, $c = 0.999999$, and $c' = 0.999999$, 0.9999, 0.99, 0.99, 0.9, 0.7, 0.5, 0.3, 0.1, and 0.0.

The results of Table 3.41 are discouraging. The poor convergence performance of both LLD and LD are not the result of negative kink-factors since we used the non-negative kink-floor option. As with the non-uniform grid problems, we examine the error following relaxation for an explanation (Figure 3.4).

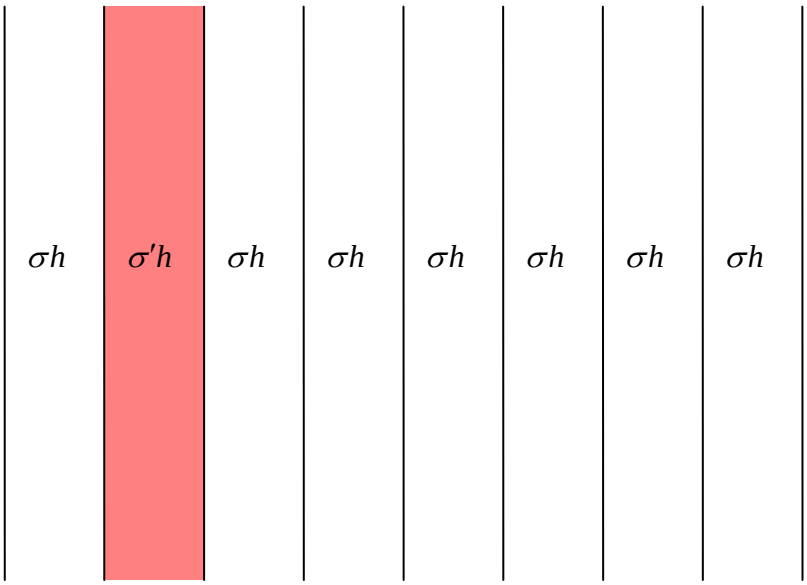

Figure 3.3 – Heterogeneous Materials & Uniform Grid Problem Suite

Table 3.41 – Predicted and Observed Convergence Ratios for Heterogeneous Test Problems; Floor ON

Scattering Ratio	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER_1D	Fourier S_2	S_NAPPER_1D
0.999999	1.1×10^{-4}	1.8×10^{-4}	1.2×10^{-4}	1.5×10^{-4}
0.9999	2.0×10^{-2}	1.0×10^{-1}	5.4×10^{-2}	0.11
0.99	0.46	0.51	0.14	0.27
0.9	0.87	0.90	0.78	0.82
0.7	0.94	0.95	0.90	0.92
0.5	0.96	0.96	0.93	0.94
0.3	0.96	0.97	0.95	0.95
0.1	0.97	0.97	0.96	0.96
0.0	0.97	0.97	0.96	0.96

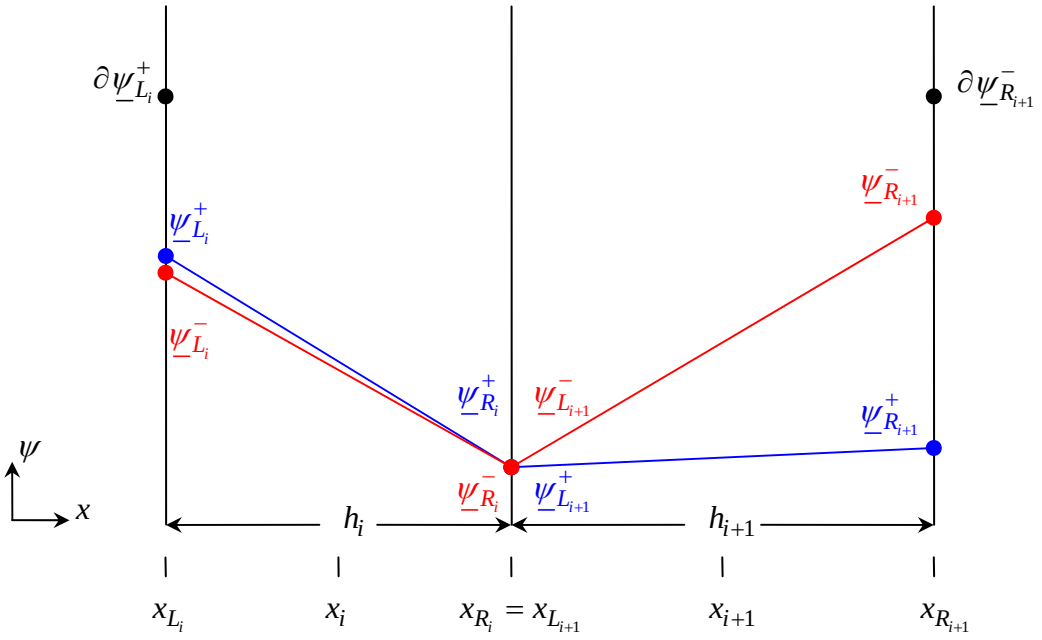

Figure 3.4 – Error Following Relaxation for a Heterogeneous Problem, $c' < 1.0$

Figure 3.4 implies one possible explanation for the poor convergence behavior. The spatial shape of the solution describing leftward moving particles has a greater deviation from linearity (*i.e.*, kink-factor) than the solution describing rightward moving particles. To test this hypothesis, we developed a version of our multigrid algorithm that allowed for two kink-factors for prolongation, one for right-moving particles and a different one for left. The Fourier analysis code is the ideal framework for testing this algorithm. As discussed in Chapter II, using more than one kink-factor has implications for the ease of constructing the two-cell inverse; however, the Fourier analysis code requires far less effort to implement this option. Further, it will tell us if it is worth the effort. Building prolongation operators with two kink-factors, we rerun the problems in Table 3.41.

Table 3.42 – Predicted and Observed Convergence Ratios for Heterogeneous Test Problems; Two Kink-Factor Method; Floor ON

Scattering Ratio	Fourier S_2, LLD ($\theta = 1.0$)		Fourier S_2, LD ($\theta = 3.0$)	
	One Kink	Two Kinks	One Kink	Two Kinks
0.999999	1.1×10^{-4}	2.7×10^{-5}	1.2×10^{-4}	1.6×10^{-6}
0.9999	2.0×10^{-2}	2.0×10^{-2}	5.4×10^{-2}	5.4×10^{-2}
0.99	0.46	0.15	0.14	0.14
0.9	0.87	0.57	0.78	0.27
0.7	0.94	0.79	0.90	0.58
0.5	0.96	0.85	0.93	0.72
0.3	0.96	0.89	0.95	0.79
0.1	0.97	0.91	0.96	0.83
0.0	0.97	0.91	0.96	0.84

The results in Table 3.42 imply that two kink-factors are probably not worth the effort. The use of two kink-factors does yield the rapidly convergent behavior observed in the homogeneous, and some of the non-uniform grid problems. The next logical extension is the use of four kink-factors – one per angular solution per cell in the S_2 case. Again, this implementation is only in the Fourier analysis code, and we repeat the problems in the Tables 3.41 and 3.42.

Table 3.43 – Predicted and Observed Convergence Ratios for Heterogeneous Test Problems; Multiple Kink-Factor Comparison; Floor ON

Scattering Ratio	Fourier S_2, LLD ($\theta = 1.0$)		Fourier S_2, LD ($\theta = 3.0$)	
	Two Kinks	Four Kinks	Two Kinks	Four Kinks
0.999999	2.7×10^{-5}	2.7×10^{-6}	1.6×10^{-6}	2.7×10^{-6}
0.9999	2.0×10^{-2}	2.6×10^{-2}	5.4×10^{-2}	5.5×10^{-2}
0.99	0.15	0.13	0.14	0.14
0.9	0.57	0.20	0.27	0.27
0.7	0.79	0.21	0.58	0.58
0.5	0.85	0.21	0.72	0.72
0.3	0.89	0.21	0.79	0.79
0.1	0.91	0.21	0.83	0.83
0.0	0.91	0.21	0.84	0.84

The results of Table 3.43 indicate that there is substantial improvement in the LLD results when using four kink-factors; however, as is the trend with this multigrid method, the same is not true for LD. In fact, almost no change is seen in the LD results, and the LLD results do not produce the rapid convergence performance that can be achieved with more traditional fast iterative methods such as Diffusion Synthetic Acceleration (DSA) [5]. The results of Tables 3.42 and 3.43 imply that increasing the complexity of the prolongation operator, and, necessarily the construction of the two-cell inverse, does not yield the desired rapid convergence rate. The ultimate conclusion of this section is that Manteuffel's error characterization following relaxation does not generalize to heterogeneous problems.

One-Dimensional Numerical Results – Heterogeneous Materials and Non-Uniform Grid Problems

The final class of problems to be considered is a set of problems with heterogeneous materials partitioned with non-uniform grids. Combinations of problems in the previous two sections are used to represent this problem class. In particular, we take halves of the most difficult problems, combine them and observe the behavior. Such a combination that proves to be particularly difficult has the cell thickness arrangement described in Table 3.44. The scattering properties are described in Table 3.45, with $c = 0.999999$.

Table 3.44 – Cell-Thickness Arrangement for Heterogeneous, Non-Uniform Problems

Problem Domain (Left-to-Right Ordering of Cell Thicknesses)							
1.0×10^{2}	1.0×10^{2}	1.0×10^{2}	1.0×10^{2}	1.0×10^{5}	1.0×10^{6}	1.0×10^{7}	1.0×10^{0}

Table 3.45 – Scattering Ratios for Heterogeneous, Non-Uniform Test Suite

Prob. #	Scattering Ratios (Left-to-Right Ordering)							
1	*c*	*c*	*c*	*c*	*c*	*c*	*c*	*c*
2	*c*	0.9999	*c*	*c*	*c*	*c*	*c*	*c*
3	*c*	0.99	*c*	*c*	*c*	*c*	*c*	*c*
4	*c*	0.9	*c*	*c*	*c*	*c*	*c*	*c*
5	*c*	0.7	*c*	*c*	*c*	*c*	*c*	*c*
6	*c*	0.5	*c*	*c*	*c*	*c*	*c*	*c*
7	*c*	0.3	*c*	*c*	*c*	*c*	*c*	*c*
8	*c*	0.1	*c*	*c*	*c*	*c*	*c*	*c*
9	*c*	0.0	*c*	*c*	*c*	*c*	*c*	*c*

Table 3.46 – Predicted and Observed Convergence Ratios for Heterogeneous, Non-Uniform Test Suite

Problem Number	LLD ($\theta = 1.0$)		LD ($\theta = 3.0$)	
	Fourier S_2	S_NAPPER	Fourier S_2	S_NAPPER
1	0.40	0.40	0.35	0.35
2	0.40	0.40	0.34	0.34
3	0.40	0.40	0.34	0.34
4	0.40	0.40	0.36	0.36
5	0.40	0.40	0.45	0.47
6	0.40	0.40	0.51	0.53
7	0.40	0.40	0.54	0.56
8	0.41	0.41	0.57	0.59
9	0.41	0.41	0.58	0.60

Unlike the other classes of problems, Table 3.46 shows that this multigrid method is insensitive to the scattering ratio when the spatial discretization is LLD. A similar degradation in performance as the heterogeneity becomes highly absorbing can be seen with LD as was the case in heterogeneous, uniform grid problems.

Eigenvalue Analysis

In addition to spectral radius computation, the Fourier analysis also allows for the characterization of all eigenvalues associated with all error modes. This type of characterization allows for the analysis of an iterative method as a preconditioner rather than an independent solver. For the problems where this multigrid method has demonstrated poor convergence behavior, we use the Fourier analysis to assess its preconditioning properties. Researchers have shown than Krylov solvers are rapidly converging if all eigenvalues associated with all error modes are well clustered, provided that there are not too many degenerate (*i.e.*, linearly dependent) eigenvectors [32-34]. To

visualize this property, we compute all eigenvalues and plot them in the real-complex plane. As previously discussed, each Fourier wave number, λ, will have thirty-two (32) eigenvalues associated with it. Of those thirty-two (32), eight (8) will be non-zero. In each of the following plots, we examine one-thousand-and-one (1,001) points in the Fourier wave number interval:

$$0 \le \lambda \le \frac{2\pi}{\sigma_t h}, \tag{3.32}$$

where, in this case, *h* represents the width of the eight-cell problem. Consequently, each plot displays eight-thousand and eight (8,008) eigenvalues – the addition eight (8) are the result of the end point being repeated. Along with the characterization of our multigrid method, we characterize the performance of Manteuffel's method as well. Recall that Manteuffel's restriction and prolongation operators are different from ours, as well as his definition of kink-factors. These differences are sufficient to question whether his method would have performed better than ours for the previous problems even though he did not thoroughly, and in some cases not-at-all, test them.

We select problems from each of the four problem classes that represent poor convergence performance. In each case, we present plots of the eigenvalue distribution in the unit disk of the complex-real plane. Further, we examine them in terms of a preconditioner. A preconditioned Richardson's iteration takes the form of Eq. (3.33):

$$\underline{\psi}^{\ell}_{(n+1)} = (I - PA)\underline{\psi}^{\ell}_{(n)} + P\underline{q}, \tag{3.33}$$

where *P* is the preconditioner. A good preconditioner will result in the inverse of the operator A being well approximated by P:

$$P \approx A^{-1}. \tag{3.34}$$

If Eq. (3.34) is true, then the convergence of Eq. (3.33) should be rapid [5]. To evaluate this method's preconditioning properties, we examine one-minus the eigenvalues of the Fourier iteration matrix. Consequently, the unit disk in the our eigenvalue analysis is centered at Re(1,0).

From the first class, homogeneous materials partitioned by a uniformly spaced grid, the poorest performance occurs for both LLD and LD with each cell ten (10) mean-

free-paths thick, and a scattering ratio of $c = 0.99$. Figure 3.5 shows that the eigenvalues associated with this problem are well clustered. This is not surprising since this was rapidly converging, thus all of the eigenvalues had to be in a small neighborhood around zero. More interesting is the identical behavior of our method and Manteuffel's method for this problem.

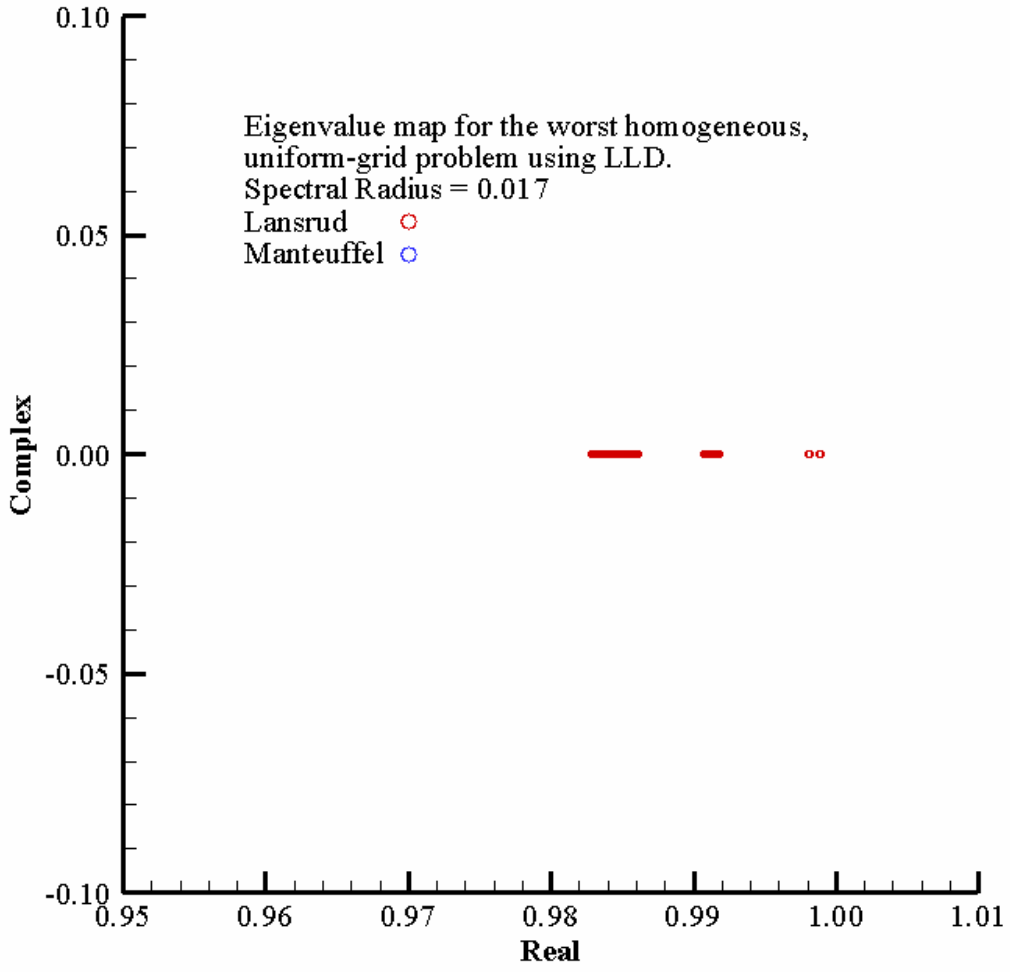

Figure 3.5 – Eigenvalue Map for the Worst Homogeneous, Uniform Problems; LLD

Figure 3.6 shows that, despite the slightly higher spectral radius, the eigenvalues are real and well clustered. As with the LLD case, the two methods are identical for the LD case.

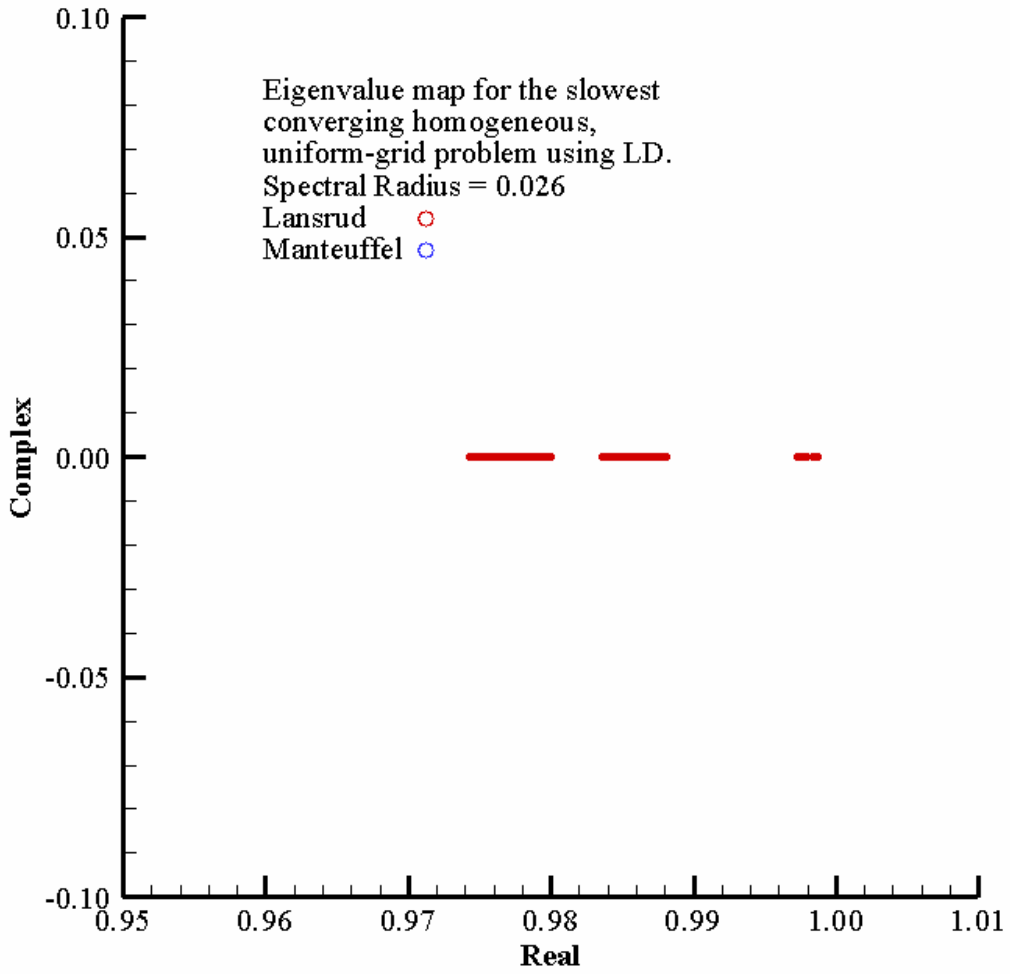

Figure 3.6 – Eigenvalue Map for the Worst Homogeneous, Uniform Problem, LD

The next problems to examine are from the homogeneous, non-uniform class. For this class of problems, the worst performance is observed, with the kink-floor ON, for problem number eight (8) with a scattering ratio of $c = 0.9999$ throughout the problem. Figure 3.7 shows slightly different behavior for the two problems. While the spectral radii are identical, our method leaves the eigenvalues better clustered than Manteuffel's. Figure 3.8 shows a slightly different behavior for the LD solution to the same problem. Spectral radii are identical, but the clustering is better for our method. Next, we examine the worst case for the heterogeneous, uniform-grid problem-class. The worst convergence behavior was observed for cells one-hundred (100) mean-free-paths thick, with a pure absorber in cell two of eight, surrounded by nearly pure scatterers.

Figure 3.9 shows that despite the spectral radius near unity, the eigenvalues are sufficiently clustered to make this iterative method a candidate for a preconditioner. Further, as with the homogeneous, uniform grid problems, the two multigrid methods perform identically. Figure 3.10 again shows identical eigenvalue clustering for the two methods; however, we now see a particularly striking set of eigenvalues near the origin. Despite its unusual shape, this still indicates good preconditioning properties

For the final class of problems, the worst behavior occurs, for both LLD and LD, during problem number nine as described in Table 3.45. Figure 3.11 shows the most striking difference between the two method. While Manteuffel's method will converge with a smaller spectral radius, our method preconditions the eigenvalues far better. Figure 3.12 again shows a our method is a more effective preconditioner than Manteuffel's. For this problem, each method has the same spectral radius, but our method preconditions leaves the eigenvalues better clustered.

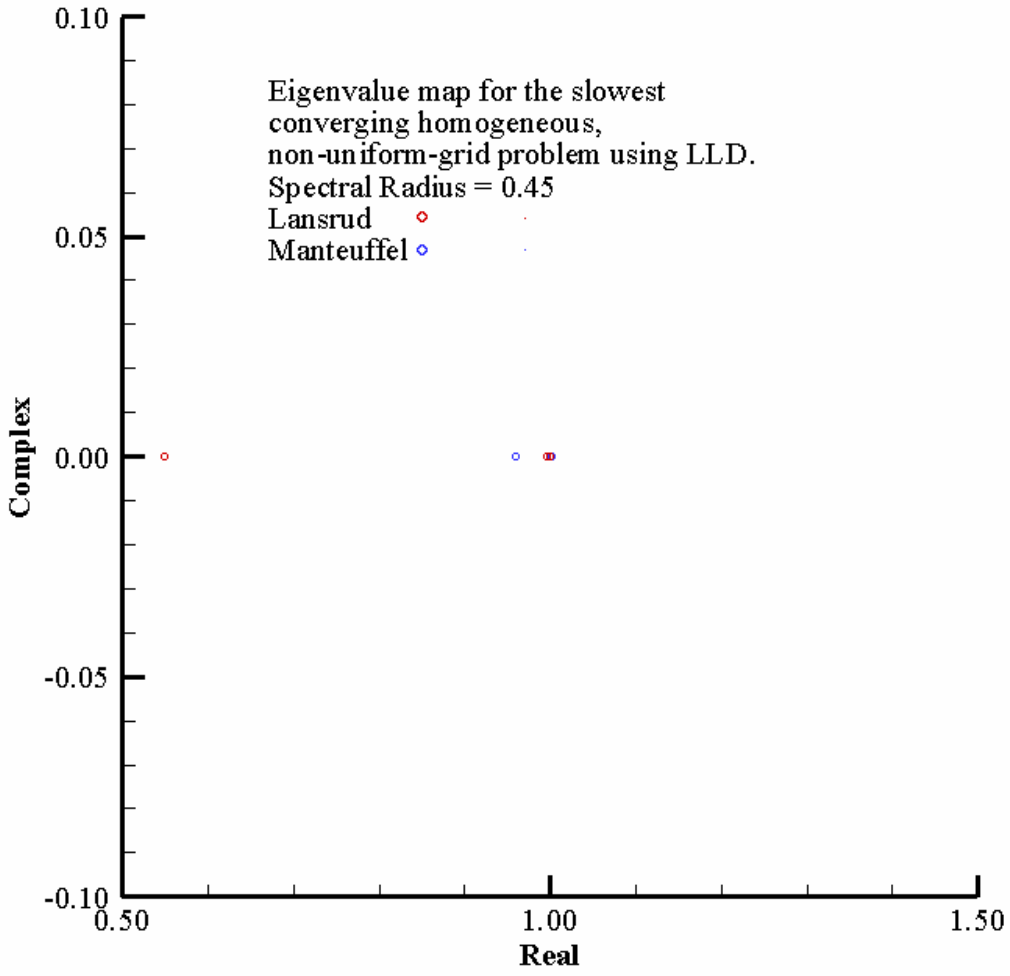

Figure 3.7 – Eigenvalue Map for the Worst Homogeneous Non-Uniform Problem, LLD

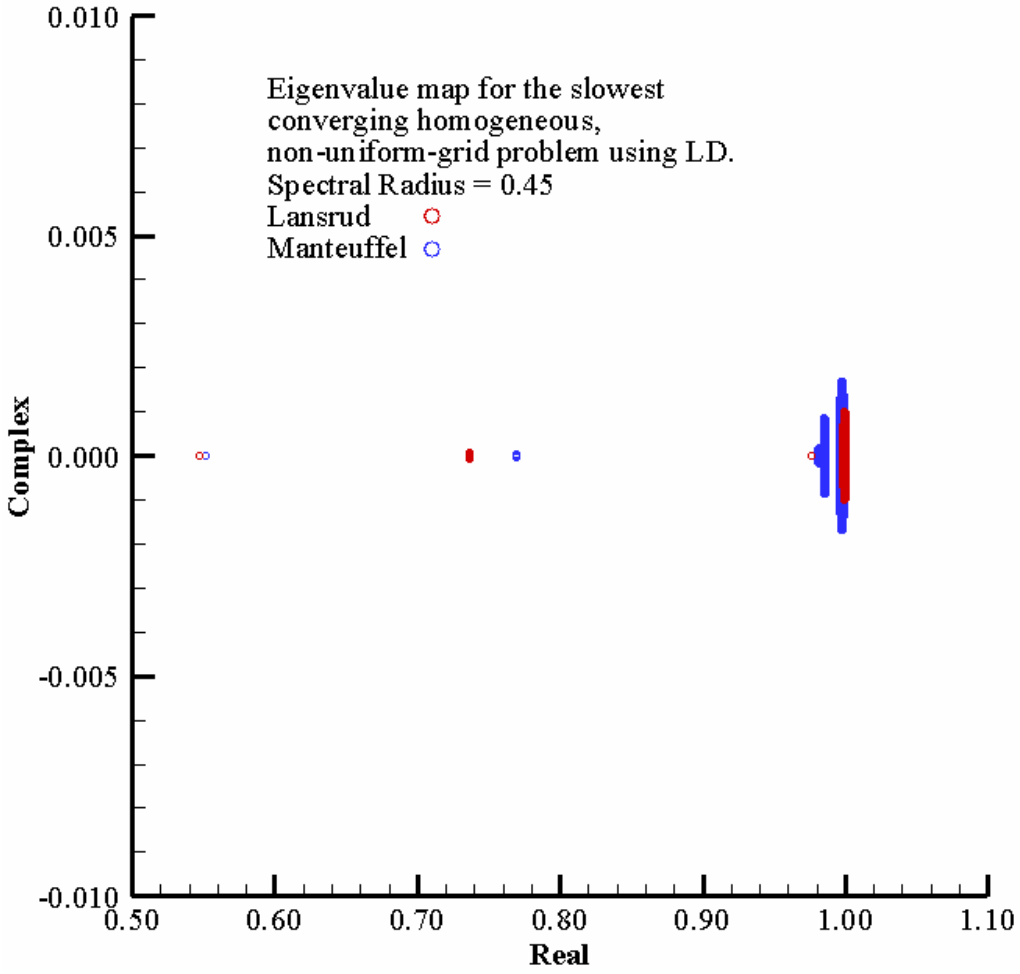

Figure 3.8 – Eigenvalue Map for the Worst Homogeneous Non-Uniform Problem, LD

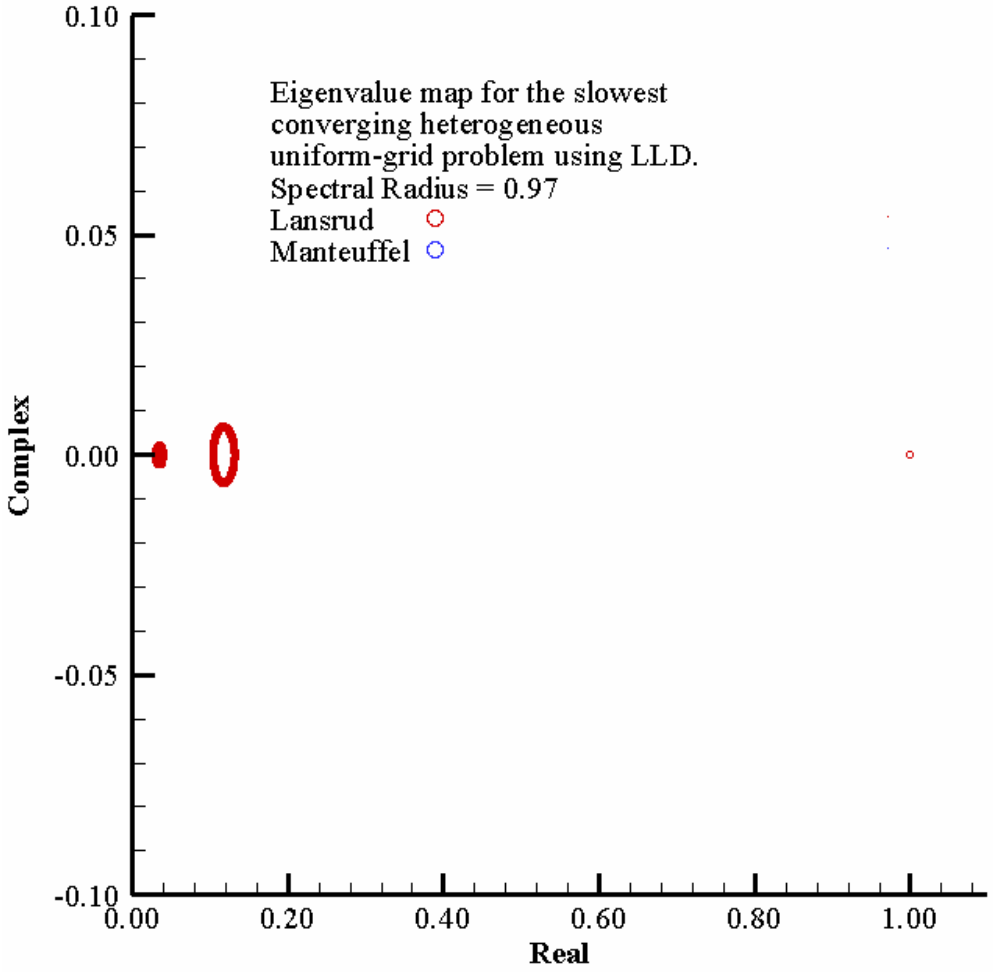

Figure 3.9 – Eigenvalue Map for the Worst Heterogeneous, Uniform Problem, LLD

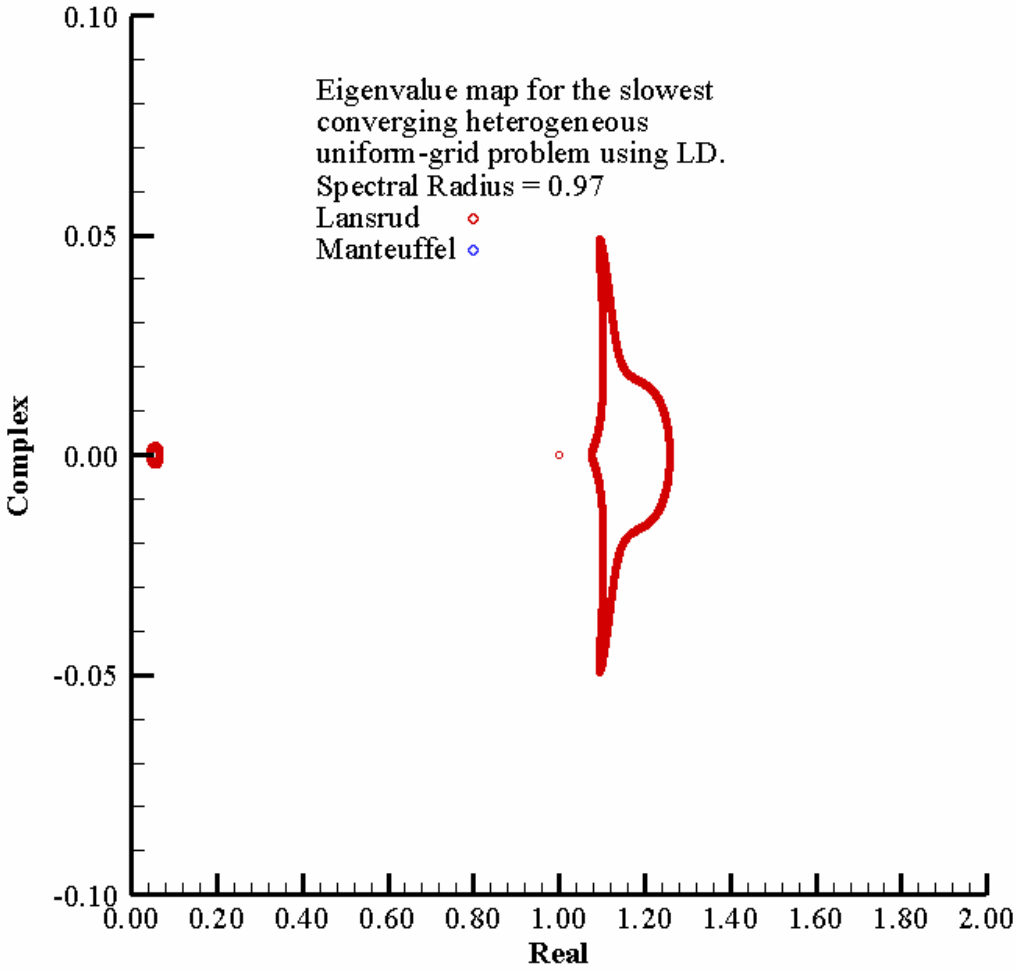

Figure 3.10 – Eigenvalue Map for the Worst Heterogeneous, Uniform Problem, LD

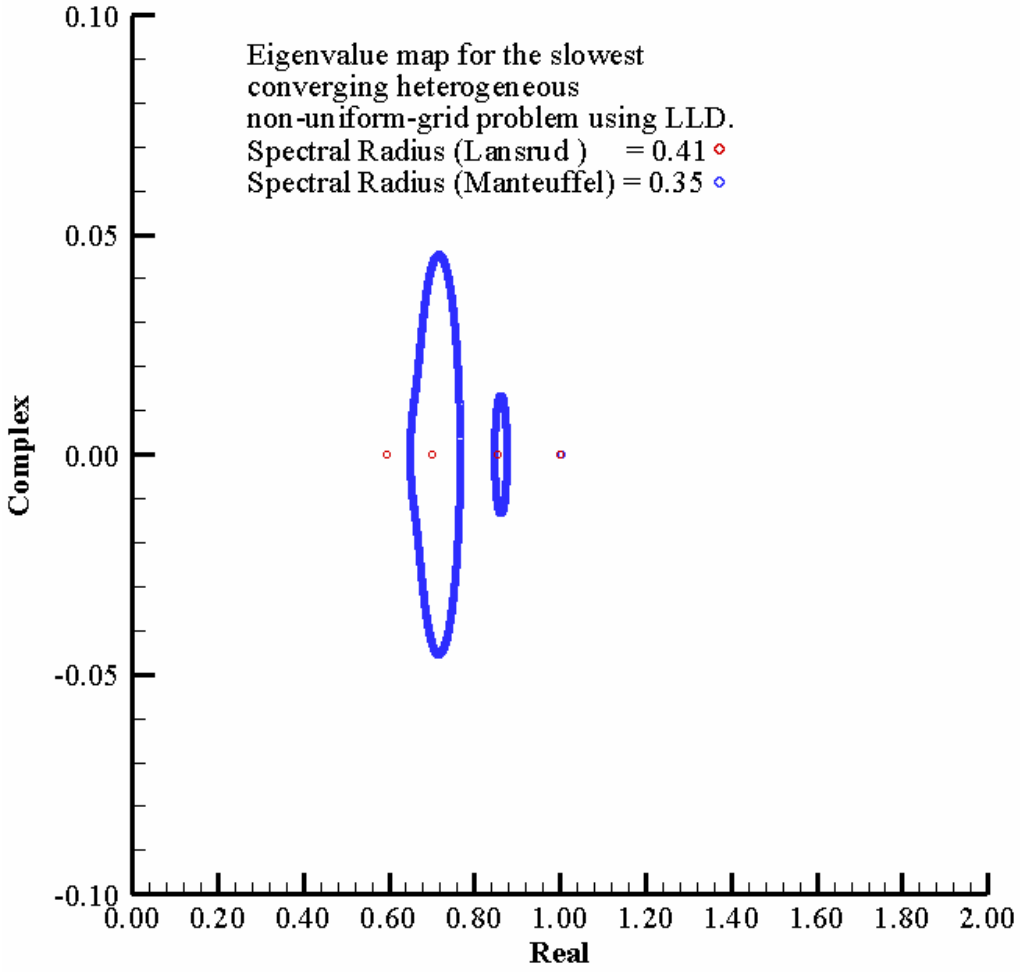

Figure 3.11 – Eigenvalue Map for the Worst Heterogeneous, Non-Uniform Problem, LLD

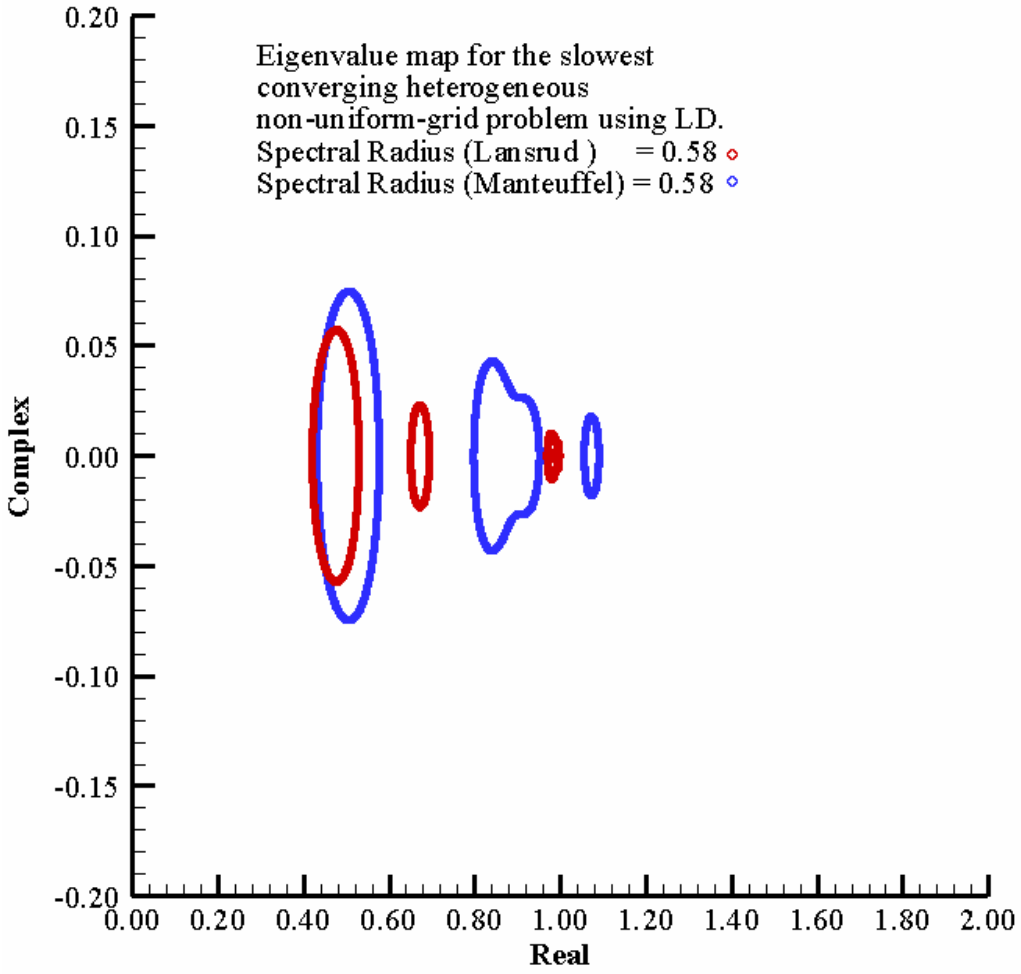

Figure 3.12 – Eigenvalue Map for the Worst Heterogeneous, Non-Uniform Problem, LD

Summary of Chapter III

In this chapter, we presented numerical results for our extension of Manteuffel's rapidly converging one-dimensional multigrid method. The results presented come from two different FORTRAN90 codes – a one-dimensional, discrete-ordinates multigrid solver (S_NAPPER_1D), and a Fourier analysis code that mimics an infinite medium and uses a three-level multigrid algorithm. While Chapter II presented analytic arguments for the behavior of this method, the results in this chapter reflect the true convergence behavior. This convergence behavior is accurately predicted by the Fourier results. Numerical results were presented by discussing four classes of one-dimensional problems. Each of the four classes, along with an eigenvalue analysis, contained a lesson. Each of these lessons learned will guide and augment the extension of this method to two-dimensions.

The first class of problems, homogeneous material partitioned by a uniform grid, confirmed the results presented by Manteuffel, *et al.* [27,28]. More importantly, this class taught two valuable lessons. First, it taught us that the shape of the error following relaxation must be characterized with operators build from fine-grid information. This shape does not necessarily reflect the LDFEM solution on coarser grids. In fact, if there is any absorption present in the problem, it will not reflect LLD very well. The second lesson, which is intimately related to the first, is that LD is a more difficult problem for this method to solve. Not only does the convergence ratio exceed that of LLD, but the error shape following relaxation is in no way related to the transport equation on coarser grids if there is any absorption present.

The second class of problems, homogenous material partitioned by a non-uniform grid, taught us that negative kink-factors can not only result in degraded performance, as with LLD, but they can lead to divergent behavior, as with LD. This problem is easily resolved by employing a lower-bound on kink-factors such that they remain non-negative. While this is effective for this class of problems, it may not be an option that should be universally applied. As was shown, if kink-factors are not allowed

to take on negative values, the rapid convergence observed for the first class of problems can be affected.

The third class of problems, heterogeneous problems partitioned by a uniform grid, shows that the convergence ratio can be made close to one. The cause of this degraded performance lies in the violation of the two tacit assumptions in the prolongation operator – isotropy and continuity. Recognition of this leads to the investigation of more than one kink-factor. For this investigation, multiple kink-factors were tested in the Fourier code. Fourier analysis of the multiple kink-factor option does not lead to significant performance improvements. Note that this modification alters the form of the two-cell operator on coarser grids. This, in turn, leads to a complication in the Sherman-Morrison decomposition that could lead to a more computationally expensive two-cell inversion step. Resolution of this question is one of our suggestions for future work.

The fourth class of problems, heterogeneous materials partitioned by non-uniform grids, is the most revealing for the eigenvalue analysis of Section VII. Evaluating this iterative method as a possible preconditioner rather than an independent solver leads to interesting results. While Manteuffel's method results in smaller spectral radii for some of the examined problems, the clustering of the eigenvalues, which implies suitability as a preconditioner, is better with our method. The addition of our Fourier analysis to the examination of this iterative method supplies an argument for this method's use as a preconditioner for a Krylov solver. With these lessons learned, we proceed to the extension of this method to two-dimensions.

CHAPTER IV

THE TWO-DIMENSIONAL PROBLEM

Introduction to Chapter IV

Chapters II and III describe a multigrid iterative method for solving one-dimensional discrete-ordinate particle transport problems present corresponding numerical results. Manteuffel's initial results were very promising; however, our work indicates that the convergence ratio of his method can approach unity in difficult heterogeneous problems. Fourier analysis suggests that, despite its shortcomings, Manteuffel's method, and our extensions to it, could be an effective preconditioner for Krylov solvers.

In this Chapter we extend Manteuffel's algorithm to more complex two-dimensional problems, employing lessons learned from our one-dimensional investigations. Section II presents a family of Bilinear Discontinuous Finite Elements (BLDFEMs) we use to discretize the spatial variable for the two-dimensional problem. Further, Section II presents the block-matrix form of the two-dimensional linear system. Section III presents our two-dimensional four-cell relaxation step. This is the extension of the one-dimensional two-cell relaxation step. As in the one-dimensional case, we consider the decomposition of the four-cell operator into the form required by Sherman-Morrison and the procedures required to form the four-cell inverse efficiently. Section IV investigates the shape of the error following relaxation by a four-cell inverse. This is more complicated that in the one-dimensional case, and we present a method for performing this characterization based on the slowest converging error mode as determined by our Fourier analysis. Section V present the construction of restriction and prolongation operators for the two-dimensional problem. These operators then define the construction of coarse-grid operators in same manner used in Chapter II for the one-dimensional case. In addition to the procedural matter of operator construction, we present a methodology for minimizing the storage requirements of this method. This was not a concern in one dimension; however, transport problems can become memory

limited in higher dimensions. We construct our method such that problems that are not too large as written will not become memory limited using this multigrid procedure.

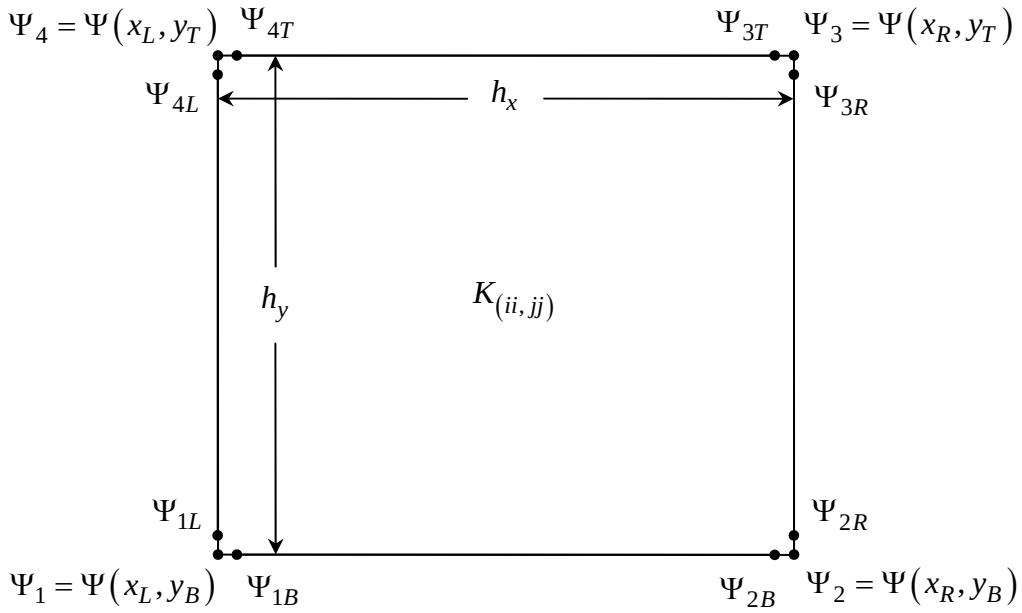

Figure 4.1 – A Rectangular Finite Element

A Family of Bilinear Discontinuous Finite Element Methods (BLDFEMs)

The two-dimensional, steady-state, mono-energetic, discrete-ordinates transport equation is given by Eq. (4.1):

$$\mu\frac{\partial\Psi_k}{\partial x}+\eta\frac{\partial\Psi_k}{\partial y}+\sigma_t(x,y)\Psi(x,y)=\frac{\sigma_s(x,y)}{4}\sum_{k'=1}^{N}w_{k'}\Psi_{k'}(x,y)+\frac{q_k(x,y)}{4}. \qquad (4.1)$$

We discretize the spatial variable on a structured, rectangular grid (Figure 4.1) with discontinuous, bilinear finite element weight and basis functions:

$$b_1(x,y)=w_1(x,y)=\left(\frac{x-x_L}{h_x}\right)\left(\frac{y-y_B}{h_y}\right), \qquad (4.2)$$

$$b_2(x,y) = w_2(x,y) = \left(\frac{x_R - x}{h_x}\right)\left(\frac{y - y_B}{h_y}\right), \tag{4.3}$$

$$b_3(x,y) = w_3(x,y) = \left(\frac{x_R - x}{h_x}\right)\left(\frac{y_T - y}{h_y}\right), \tag{4.4}$$

$$b_4(x,y) = w_4(x,y) = \left(\frac{x - x_L}{h_x}\right)\left(\frac{y_T - y}{h_y}\right). \tag{4.5}$$

Equations (4.2) - (4.5) represent cardinal weight and basis functions. Each are unity at its support point and zero at all other support points. This selection of weight and basis functions guarantees certain desirable properties in optically thick, diffusive problems – robustness, accuracy, and reasonable behavior in the presence of unresolved boundary layers [35].

Multiplying Eq. (4.1), by Eqs. (4.2) - (4.5), and integrating over the cell, we obtain the standard bilinear discontinuous (BLD) equations. Introducing the lumping parameters $\theta^x_{K_{(ii,jj)}}$ and $\theta^y_{K_{(ii,jj)}}$ into this system of equations, we obtain a family of BLD equations defined by Eq. (4.6):

$$\begin{aligned}
&\mu\left(\frac{h^y_{K_{(ii,jj)}}}{2}\right)\partial\Psi^y_{K_{(ii,jj)}} + \eta\left(\frac{h^x_{K_{(ii,jj)}}}{2}\right)\partial\Psi^x_{K_{(ii,jj)}} + \mu\left(\frac{h^y_{K_{(ii,jj)}}}{4}\right)\mathrm{M}_{K_{(ii,jj)}}\Psi_{K_{(ii,jj)}} + \\
&\eta\left(\frac{h^x_{K_{(ii,jj)}}}{4}\right)\mathrm{N}_{K_{(ii,jj)}}\Psi_{K_{(ii,jj)}} + \left(\frac{\left(\sigma^t_{K_{(ii,jj)}}\right)\left(h^x_{K_{(ii,jj)}}\right)\left(h^y_{K_{(ii,jj)}}\right)}{4}\right)\mathrm{T}_{K_{(ii,jj)}}\Psi_{K_{(ii,jj)}} - \\
&\left(\frac{\left(\sigma^s_{K_{(ii,jj)}}\right)\left(h^x_{K_{(ii,jj)}}\right)\left(h^y_{K_{(ii,jj)}}\right)}{4(4\pi)}\right)\mathrm{T}_{K_{(ii,jj)}}\Theta_{K_{(ii,jj)}} = \left(\frac{\left(h^x_{K_{(ii,jj)}}\right)\left(h^y_{K_{(ii,jj)}}\right)}{4(4\pi)}\right)\mathrm{T}_{K_{(ii,jj)}}Q_{K_{(ii,jj)}}.
\end{aligned} \tag{4.6}$$

The individual vectors, operators and coefficients are defined by Eqs. (4.7) - (4.18):

$$\partial\Psi^{y}_{K_{(ii,jj)}} = \begin{bmatrix} -\left(\frac{\theta^{y}_{K_{ii,jj}}+1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{1L}_{K_{(ii,jj)}} - \left(\frac{\theta^{y}_{K_{ii,jj}}-1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{4L}_{K_{(ii,jj)}} \\ \left(\frac{\theta^{y}_{K_{ii,jj}}+1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{2R}_{K_{(ii,jj)}} + \left(\frac{\theta^{y}_{K_{ii,jj}}-1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{3R}_{K_{(ii,jj)}} \\ \left(\frac{\theta^{y}_{K_{ii,jj}}-1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{2R}_{K_{(ii,jj)}} + \left(\frac{\theta^{y}_{K_{ii,jj}}+1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{3R}_{K_{(ii,jj)}} \\ -\left(\frac{\theta^{y}_{K_{ii,jj}}-1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{1L}_{K_{(ii,jj)}} - \left(\frac{\theta^{y}_{K_{ii,jj}}+1}{2\theta^{y}_{K_{ii,jj}}}\right)\psi^{4L}_{K_{(ii,jj)}} \end{bmatrix}, \tag{4.7}$$

$$\partial\Psi^{x}_{K_{(ii,jj)}} = \begin{bmatrix} -\left(\frac{\theta^{x}_{K_{ii,jj}}+1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{1B}_{K_{(ii,jj)}} - \left(\frac{\theta^{x}_{K_{ii,jj}}-1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{2B}_{K_{K_{(ii,jj)}}} \\ -\left(\frac{\theta^{x}_{K_{ii,jj}}-1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{1B}_{K_{(ii,jj)}} - \left(\frac{\theta^{x}_{K_{ii,jj}}+1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{2B}_{K_{K_{(ii,jj)}}} \\ \left(\frac{\theta^{x}_{K_{ii,jj}}+1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{3T}_{K_{K_{(ii,jj)}}} + \left(\frac{\theta^{x}_{K_{ii,jj}}-1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{4T}_{K_{K_{(ii,jj)}}} \\ \left(\frac{\theta^{x}_{K_{ii,jj}}-1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{3T}_{K_{K_{(ii,jj)}}} + \left(\frac{\theta^{x}_{K_{ii,jj}}+1}{2\theta^{x}_{K_{ii,jj}}}\right)\psi^{4T}_{K_{K_{(ii,jj)}}} \end{bmatrix}, \tag{4.8}$$

$$\mathrm{M}_{K_{(ii,jj)}} = \begin{bmatrix} \left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) \\ -\left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) \\ -\left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) \\ \left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right) \end{bmatrix}, \tag{4.9}$$

$$\mathrm{N}_{K_{(ii,jj)}} = \begin{bmatrix} \left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) \\ \left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & \left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) \\ -\left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) \\ -\left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right) & -\left(\frac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right) \end{bmatrix}, \tag{4.10}$$

$$\mathrm{T}_{K_{(ii,jj)}} = \begin{bmatrix} \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} \\ \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} \\ \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} \\ \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} \end{bmatrix} \tag{4.11}$$

$$\left(\theta_{xy}^{++}\right)_{K_{(ii,jj)}} = \left(\frac{\theta^x_{K_{(ii,jj)}}+1}{2\theta^x_{K_{(ii,jj)}}}\right)\left(\frac{\theta^y_{K_{(ii,jj)}}+1}{2\theta^y_{K_{(ii,jj)}}}\right), \tag{4.12}$$

$$\left(\theta_{xy}^{-+}\right)_{K_{(ii,jj)}} = \left(\frac{\theta^x_{K_{(ii,jj)}}-1}{2\theta^x_{K_{(ii,jj)}}}\right)\left(\frac{\theta^y_{K_{(ii,jj)}}+1}{2\theta^y_{K_{(ii,jj)}}}\right), \tag{4.13}$$

$$\left(\theta_{xy}^{--}\right)_{K_{(ii,jj)}} = \left(\frac{\theta^x_{K_{(ii,jj)}}-1}{2\theta^x_{K_{(ii,jj)}}}\right)\left(\frac{\theta^y_{K_{(ii,jj)}}-1}{2\theta^y_{K_{(ii,jj)}}}\right), \tag{4.14}$$

$$\left(\theta_{xy}^{+-}\right)_{K_{(ii,jj)}} = \left(\frac{\theta^x_{K_{(ii,jj)}}+1}{2\theta^x_{K_{(ii,jj)}}}\right)\left(\frac{\theta^y_{K_{(ii,jj)}}-1}{2\theta^y_{K_{(ii,jj)}}}\right), \tag{4.15}$$

$$\Psi_{K_{(ii,jj)}} = \left[\Psi^1_{K_{(ii,jj)}} \quad \Psi^2_{K_{(ii,jj)}} \quad \Psi^3_{K_{(ii,jj)}} \quad \Psi^4_{K_{(ii,jj)}}\right]^T, \tag{4.16}$$

$$\Theta_{K_{ii,jj}} = \left[\sum_{kk=1}^{N} w_{kk}\Psi_{K^1_{(ii,jj)},kk} \quad \sum_{kk=1}^{N} w_{kk}\Psi_{K^2_{(ii,jj)},kk} \quad \sum_{kk=1}^{N} w_{kk}\Psi_{K^3_{(ii,jj)},kk} \quad \sum_{kk=1}^{N} w_{kk}\Psi_{K^4_{(ii,jj)},kk}\right]^T \tag{4.17}$$

and:

$$Q_{K_{ii,jj}} = \left[Q_{K^1_{(ii,jj)}} \quad Q_{K^2_{(ii,jj)}} \quad Q_{K^3_{(ii,jj)}} \quad Q_{K^4_{(ii,jj)}}\right]^T. \tag{4.18}$$

Equations (4.6) - (4.18) describe the finite-element solution for a single cell, $[ii, jj]$, as a function of a single discrete ordinate, $\omega_k = \mu\vec{i} + \eta\vec{j}$. To ensure that our method is as generally applicable as possible, we allow the quadrature set to be asymmetric, with different numbers of points in each directional quadrant. Construction of such a quadrature set is beyond the scope of this work; however, our notation and methodology admit the possibility of its use (Figure 4.2).

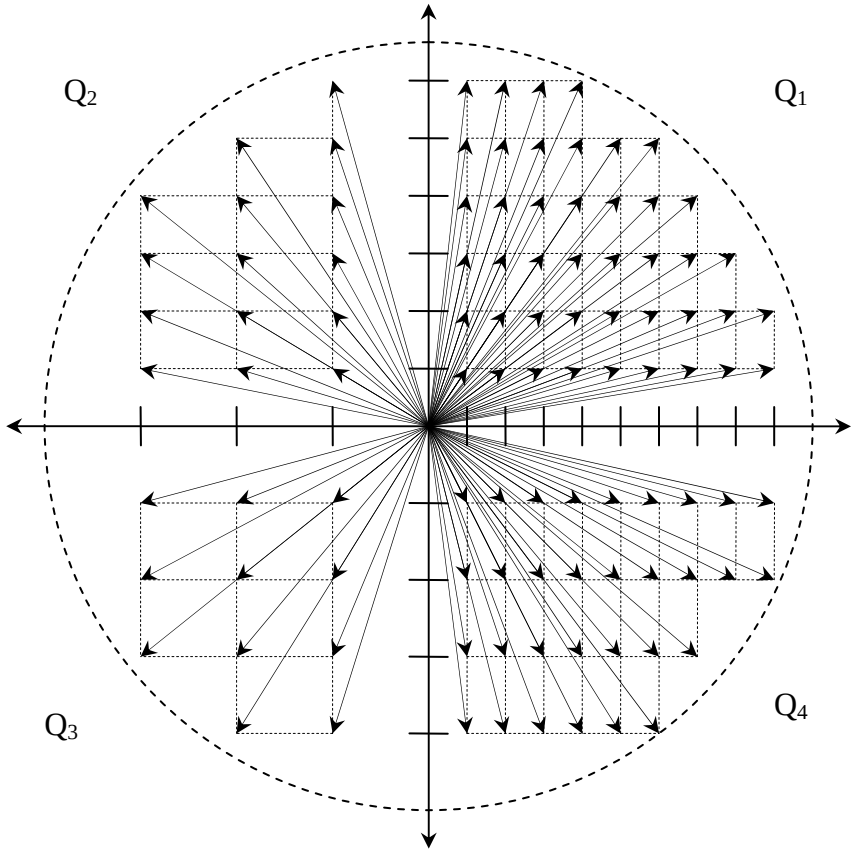

Figure 4.2 – Visualization of an Asymmetric Quadrature Set

We abstract the S_N problem to allow each of the angular quadrants, Q_i, to have a different number of discrete-ordinates, N_i, each of the form $\omega_k = \mu_k \vec{i} + \eta_k \vec{j}$, defined by Eq. (4.19):

$$\omega_k \in \{\omega_{kk}\}_{kk=1}^{N_i}, i = 1,2,3,4 . \tag{4.19}$$

The asymmetric S_N solution vector for a single cell, at grid level ℓ, is given by Eq. (4.20):

$$\underline{\Psi}^{\ell}_{K_{(ii,jj)}} = \left[\underline{\Psi}^{(4N_1)}_{K_{(ii,jj)}} \quad \underline{\Psi}^{(4N_2)}_{K_{(ii,jj)}} \quad \underline{\Psi}^{(4N_3)}_{K_{(ii,jj)}} \quad \underline{\Psi}^{(4N_4)}_{K_{(ii,jj)}} \right]^T , \tag{4.20}$$

where, in each quadrant, we construct the solution along each discrete-ordinate at each vertex in Eq. (4.21):

$$\underline{\Psi}^{(4N_i)}_{K_{(ii,jj)}} = \left[\underline{\Psi}^{(N_i)}_{K^1_{(ii,jj)}} \quad \underline{\Psi}^{(N_i)}_{K^2_{(ii,jj)}} \quad \underline{\Psi}^{(N_i)}_{K^3_{(ii,jj)}} \quad \underline{\Psi}^{(N_i)}_{K^4_{(ii,jj)}} \right]^T, \; i = 1,2,3,4. \tag{4.21}$$

The term that suffers the greatest complexity from the introduction of an asymmetric quadrature set is the scalar flux term, originally presented in terms of a symmetric quadrature set in Eq. (4.17). The initial extension to an asymmetric representation is straightforward. The scalar flux at each vertex is represented as four quadrature sums over each of the four angular quadrants in Eq. (4.22):

$$\Theta_{K_{ii,jj}} = \begin{bmatrix} \sum_{kk=1}^{N_1} w_{kk} \Psi_{K^1_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll} \Psi_{K^1_{(ii,jj)},ll} + \sum_{mm=1}^{N_3} w_{mm} \Psi_{K^1_{(ii,jj)},mm} + \sum_{nn=1}^{N_4} w_{nn} \Psi_{K^1_{(ii,jj)},nn} \\ \sum_{kk=1}^{N_1} w_{kk} \Psi_{K^2_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll} \Psi_{K^2_{(ii,jj)},ll} + \sum_{mm=1}^{N_3} w_{mm} \Psi_{K^2_{(ii,jj)},mm} + \sum_{nn=1}^{N_4} w_{nn} \Psi_{K^2_{(ii,jj)},nn} \\ \sum_{kk=1}^{N_1} w_{kk} \Psi_{K^3_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll} \Psi_{K^3_{(ii,jj)},ll} + \sum_{mm=1}^{N_3} w_{mm} \Psi_{K^3_{(ii,jj)},mm} + \sum_{nn=1}^{N_4} w_{nn} \Psi_{K^3_{(ii,jj)},nn} \\ \sum_{kk=1}^{N_1} w_{kk} \Psi_{K^4_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll} \Psi_{K^4_{(ii,jj)},ll} + \sum_{mm=1}^{N_3} w_{mm} \Psi_{K^4_{(ii,jj)},mm} + \sum_{nn=1}^{N_4} w_{nn} \Psi_{K^4_{(ii,jj)},nn} \end{bmatrix}. \tag{4.22}$$

To cast Eq. (4.22) in a form that will decompose into the Sherman-Morrison decomposition, we must write it as a product of quadrature weights and the solution vector. The terms needed to decompose the scalar flux operator are given in Eqs. (4.23) - (4.25):

$$\Theta_{K_{(ii,jj)}} = \left[W^{(4\times N_1)} \quad W^{(4\times N_2)} \quad W^{(4\times N_3)} \quad W^{(4\times N_4)} \right] \begin{bmatrix} \underline{\Psi}^{(N_1)}_{K_{(ii,jj)}} \\ \underline{\Psi}^{(N_2)}_{K_{(ii,jj)}} \\ \underline{\Psi}^{(N_3)}_{K_{(ii,jj)}} \\ \underline{\Psi}^{(N_4)}_{K_{(ii,jj)}} \end{bmatrix}, \tag{4.23}$$

where:

$$W^{(4\times N_i)} = \begin{bmatrix} \underline{w}^T_{(N_i)} & 0 & 0 & 0 \\ 0 & \underline{w}^T_{(N_i)} & 0 & 0 \\ 0 & 0 & \underline{w}^T_{(N_i)} & 0 \\ 0 & 0 & 0 & \underline{w}^T_{(N_i)} \end{bmatrix}, i = 1,2,3,4\,, \tag{4.24}$$

recall:

$$\underline{w}^T_{(N_i)} = \begin{bmatrix} w_1 & w_2 & \cdots & w_{N_i} \end{bmatrix}. \tag{4.25}$$

The result of the operation defined in Eq. (4.23) is a vector of length four (4) with each entry corresponding to the scalar flux at one of the vertices. This is suitable for the calculation of the angular flux for a single discrete ordinate; however, we must cast these equations to describe the entire solution vector. This is achieved by scaling each scalar flux by a unit column vector in Eq. (4.26):

$$\underline{\Theta}_{K_{(ii,jj)}} = \begin{bmatrix} W^{(N_1\times N_1)} & W^{(N_1\times N_2)} & W^{(N_1\times N_3)} & W^{(N_1\times N_4)} \\ W^{(N_2\times N_1)} & W^{(N_2\times N_2)} & W^{(N_2\times N_3)} & W^{(N_2\times N_4)} \\ W^{(N_3\times N_1)} & W^{(N_3\times N_2)} & W^{(N_3\times N_3)} & W^{(N_3\times N_4)} \\ W^{(N_4\times N_1)} & W^{(N_4\times N_2)} & W^{(N_4\times N_3)} & W^{(N_4\times N_4)} \end{bmatrix} \begin{bmatrix} \underline{\Psi}^{(N_1)}_{K_{(ii,jj)}} \\ \underline{\Psi}^{(N_2)}_{K_{(ii,jj)}} \\ \underline{\Psi}^{(N_3)}_{K_{(ii,jj)}} \\ \underline{\Psi}^{(N_4)}_{K_{(ii,jj)}} \end{bmatrix}, \tag{4.26}$$

Where the individual blocks in the previous expression are defined in Eq. (4.27):

$$W^{(N\times N')} = \begin{bmatrix} \underline{1}^{(N)}\underline{w}^T_{(N')} & 0 & 0 & 0 \\ 0 & \underline{1}^{(N)}\underline{w}^T_{(N')} & 0 & 0 \\ 0 & 0 & \underline{1}^{(N)}\underline{w}^T_{(N')} & 0 \\ 0 & 0 & 0 & \underline{1}^{(N)}\underline{w}^T_{(N')} \end{bmatrix}. \tag{4.27}$$

The final step in casting the scalar flux term in the proper form is to operate on it from the left by the appropriately scaled mass matrix. This operation is defined by Eq. (4.28)

$$\underline{\Theta}_{K_{(ii,jj)}} = \begin{bmatrix} R^{(4N_1\times 4N_1)}_{K_{(ii,jj)}} & R^{(4N_1\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_1\times 4N_3)}_{K_{(ii,jj)}} & R^{(4N_1\times 4N_4)}_{K_{(ii,jj)}} \\ R^{(4N_2\times 4N_1)}_{K_{ii(ii,jj),jj}} & R^{(4N_2\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_2\times 4N_3)}_{K_{(ii,jj)}} & R^{(4N_2\times 4N_4)}_{K_{(ii,jj)}} \\ R^{(4N_3\times 4N_1)}_{K_{(ii,jj)}} & R^{(4N_3\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_3\times 4N_3)}_{K_{(ii,jj)}} & R^{(4N_3\times 4N_4)}_{K_{(ii,jj)}} \\ R^{(4N_4\times 4N_1)}_{K_{(ii,jj)}} & R^{(4N_4\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_4\times 4N_3)}_{K_{(ii,jj)}} & R^{(4N_4\times 4N_4)}_{K_{(ii,jj)}} \end{bmatrix}. \tag{4.28}$$

The individual blocks in the previous expression are defined in Eqs. (4.29) and (4.30):

$$R^{(4N\times 4N')}_{K_{(ii,jj)}} = \begin{bmatrix} \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} R \\ \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} R \\ \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} R \\ \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} R & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} R \end{bmatrix}, \tag{4.29}$$

$$R = R^{(N\times N')} = \underline{1}^{(N)}\,\underline{w}^{T}_{(N')}. \tag{4.30}$$

With the scalar flux operator defined for the block-matrix form of the complete S_N problem, the remaining terms are straight forward. The only remaining subtlety is how the boundary operators are treated; however, this is not complicated and is left to the reader to understand that it is simply moving known (*i.e.*, boundary) information to the right-hand-side of the transport equation with the fixed-sourced information. The complete S_N transport equation in block-matrix form for a single cell is given by Eq. (4.31):

$$\left[\mathrm{M}^{\ell}_{K_{(ii,jj)}} + \mathrm{N}^{\ell}_{K_{(ii,jj)}} + \mathrm{T}^{\ell}_{K_{(ii,jj)}} - R^{\ell}_{K_{(ii,jj)}}\right]\underline{\Psi}^{\ell}_{K_{(ii,jj)}} = \partial\underline{\Psi}^{\ell}_{K_{(ii,jj)}} + \underline{Q}^{\ell}_{K_{(ii,jj)}}. \tag{4.31}$$

Recall that our multigrid method uses equations for the angular flux correction, and that these equations are simply Eq. (4.31) without the fixed-source term:

$$\left[\mathrm{M}^{\ell}_{K_{(ii,jj)}} + \mathrm{N}^{\ell}_{K_{(ii,jj)}} + \mathrm{T}^{\ell}_{K_{(ii,jj)}} - R^{\ell}_{K_{(ii,jj)}}\right]\underline{\psi}^{\ell}_{K_{(ii,jj)}} = \partial\underline{\psi}^{\ell}_{K_{(ii,jj)}}. \tag{4.32}$$

The streaming terms ($\mathrm{M}^{\ell}_{K_{(ii,jj)}}$, and $\mathrm{N}^{\ell}_{K_{(ii,jj)}}$) and the boundary-source term ($\partial\underline{\psi}^{\ell}_{K_{(ii,jj)}}$) in Eq. (4.32) are angular-quadrant dependant. The collision-rate-density and scattering-rate-density terms ($\mathrm{T}^{\ell}_{K_{(ii,jj)}}$, and $R^{\ell}_{K_{(ii,jj)}}$, respectively) are not. Since these are more easily defined, we detail them first. The collision-rate-density matrix is defined by Eq. (4.33):

$$\mathrm{T}^{\ell}_{K_{(ii,jj)}} = \frac{\left(\sigma^{t}_{K_{(ii,jj)}}\right)\left(h^{x}_{K_{(ii,jj)}}\right)\left(h^{y}_{K_{(ii,jj)}}\right)}{4}\begin{bmatrix} \mathrm{T}^{(4N_1)^2}_{K_{(ii,jj)}} & 0 & 0 & 0 \\ 0 & \mathrm{T}^{(4N_2)^2}_{K_{(ii,jj)}} & 0 & 0 \\ 0 & 0 & \mathrm{T}^{(4N_3)^2}_{K_{(ii,jj)}} & 0 \\ 0 & 0 & 0 & \mathrm{T}^{(4N_4)^2}_{K_{(ii,jj)}} \end{bmatrix}, \tag{4.33}$$

where the block-diagonal terms are given by Eq. (4.34):

$$\mathrm{T}^{(4N_k\times 4N_k)}_{K_{(ii,jj)}} = \begin{bmatrix} \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{+-}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{--}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-+}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{++}_{xy}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \end{bmatrix}, \tag{4.34}$$

for $k = 1,2,3,4$ (for brevity, we have written $I^{(N_k\times N_k)}$ as $I^{(N_k)^2}$ in Eq. (4.34) and in several equation that follow).

The scattering-rate-density term is given by Eq. (4.35) (note that the quadrature weights have been normalized so they sum to four (4)):

$$R^{\ell}_{K_{(ii,jj)}} =$$

$$\frac{\left(\sigma^{t}_{K_{(ii,jj)}}\right)\left(h^{x}_{K_{(ii,jj)}}\right)\left(h^{y}_{K_{(ii,jj)}}\right)}{16}\begin{bmatrix} R^{(4N_1\times 4N_1)}_{K_{(ii,jj)}} & R^{(4N_1\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_1\times 4N_3)}_{K_{(ii,jj)}} & R^{(4N_1\times 4N_4)}_{K_{(ii,jj)}} \\ R^{(4N_2\times 4N_1)}_{K_{(ii,jj)}} & R^{(4N_2\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_2\times 4N_3)}_{K_{(ii,jj)}} & R^{(4N_2\times 4N_4)}_{K_{(ii,jj)}} \\ R^{(4N_3\times 4N_1)}_{K_{(ii,jj)}} & R^{(4N_3\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_3\times 4N_3)}_{K_{(ii,jj)}} & R^{(4N_3\times 4N_4)}_{K_{(ii,jj)}} \\ R^{(4N_4\times 4N_1)}_{K_{(ii,jj)}} & R^{(4N_4\times 4N_2)}_{K_{(ii,jj)}} & R^{(4N_4\times 4N_3)}_{(ii,jj)} & R^{(4N_4\times 4N_4)}_{K_{(ii,jj)}} \end{bmatrix}, \quad (4.35)$$

where the individual blocks in Eq. (4.35) have been defined in Eq. (4.29).

Now we define, a quadrant at a time, the horizontal component of the streaming term starting with Eq. (4.36):

$$\mathrm{M}^{\ell}_{K_{(ii,jj)}} =$$

$$\left(\frac{h^{y}_{K_{(ii,jj)}}}{4}\right)\begin{bmatrix} \mathrm{M}^{+}_{K_{(ii,jj)}}\mathrm{M}^{(4N_1)^2} & 0 & 0 & 0 \\ 0 & \mathrm{M}^{-}_{K_{(ii,jj)}}\mathrm{M}^{(4N_2)^2} & 0 & 0 \\ 0 & 0 & \mathrm{M}^{-}_{K_{(ii,jj)}}\mathrm{M}^{(4N_3)^2} & 0 \\ 0 & 0 & 0 & \mathrm{M}^{+}_{K_{(ii,jj)}}\mathrm{M}^{(4N_4)^2} \end{bmatrix}, \quad (4.36)$$

where the individual blocks corresponding to the right-half of discrete-ordinates space are given by Eq. (4.37):

$$\left(\mathrm{M}^{+}_{K_{(ii,jj)}}\right)^{(4N_k\times 4N_k)} =$$

$$\begin{bmatrix} \left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ -\left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ -\left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{-}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{+}_{y}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \end{bmatrix}, \quad (4.37)$$

where $k = 1,4$. The individual blocks corresponding to the left-half of discrete-ordinates space are given by Eq. (4.38):

$$\left(\mathrm{M}^{-}_{K_{(ii,jj)}}\right)^{\left(4N_k \times 4N_k\right)} =$$
$$\begin{bmatrix} \left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & -\left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & -\left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} \\ \left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} \\ \left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} \\ \left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & -\left(\theta_y^-\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & -\left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} & \left(\theta_y^+\right)_{K_{(ii,jj)}} I^{\left(N_k\right)^2} \end{bmatrix}, \quad (4.38)$$

where $k = 2,3$. The other blocks in Eq. (4.36) simply list the horizontal components of the quadrant-specific, absolute-value of the directional cosines along the diagonal as shown by Eq. (4.39):

$$\mathrm{M}^{\left(N_k \times N_k\right)} = \begin{bmatrix} \left|\mu_1\right| & 0 & \cdots & 0 & 0 \\ 0 & \left|\mu_2\right| & 0 & 0 & 0 \\ \vdots & 0 & \ddots & 0 & \vdots \\ 0 & 0 & 0 & \left|\mu_{k-1}\right| & 0 \\ 0 & 0 & \cdots & 0 & \left|\mu_k\right| \end{bmatrix}, \quad (4.39)$$

where $k = 1,2,3,4$. It should be noted that the need to specify the streaming operator by half-space comes from a choice that was made during the original derivation. If the block-matrix form were derived and the signs of the directional cosines in Eq. (4.39) were left unchanged, then the horizontal streaming operator component listed in Eqs. (4.37) and (4.38) would be identical; however, this choice would also make the source terms less explicit, as we will see shortly.

The vertical component of the streaming term takes on a similar form beginning with Eq. (4.40):

$$\mathrm{N}^{\ell}_{K_{(ii,jj)}} =$$

$$\left(\frac{h^{x}_{K_{(ii,jj)}}}{4}\right)\begin{bmatrix} \mathrm{N}^{+}_{K_{(ii,jj)}}\mathrm{N}^{(4N_1)^2} & 0 & 0 & 0 \\ 0 & \mathrm{N}^{+}_{K_{(ii,jj)}}\mathrm{N}^{(4N_2)^2} & 0 & 0 \\ 0 & 0 & \mathrm{N}^{-}_{K_{(ii,jj)}}\mathrm{N}^{(4N_3)^2} & 0 \\ 0 & 0 & 0 & \mathrm{N}^{-}_{K_{(ii,jj)}}\mathrm{N}^{(4N_4)^2} \end{bmatrix}, \quad (4.40)$$

where the block-diagonal terms corresponding to the upper-half of discrete-ordinates space are given by Eq. (4.41):

$$\left(\mathrm{N}^{+}_{K_{(ii,jj)}}\right)^{(4N_k\times 4N_k)} =$$

$$\begin{bmatrix} \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ -\left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ -\left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \end{bmatrix}, \quad (4.41)$$

where $k = 1,2$. The block-diagonal terms corresponding to the lower-half of discrete-ordinates space are given by Eq. (4.42):

$$\left(\mathrm{N}^{-}_{K_{(ii,jj)}}\right)^{(4N_k\times 4N_k)} =$$

$$\begin{bmatrix} \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & -\left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \\ \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{-}\right)_{K_{(ii,jj)}} I^{(N_k)^2} & \left(\theta^{x}_{+}\right)_{K_{(ii,jj)}} I^{(N_k)^2} \end{bmatrix}, \quad (4.42)$$

where $k = 3,4$. The other blocks in Eq. (4.40) simply list the vertical components of the quadrant-specific, absolute-value of the directional cosines along the diagonal as shown by Eq. (4.43):

$$\mathrm{N}^{(N_k \times N_k)} = \begin{bmatrix} |\eta_1| & 0 & \cdots & 0 & 0 \\ 0 & |\eta_2| & 0 & 0 & 0 \\ \vdots & 0 & \ddots & 0 & \vdots \\ 0 & 0 & 0 & |\eta_{k-1}| & 0 \\ 0 & 0 & \cdots & 0 & |\eta_k| \end{bmatrix}, \tag{4.43}$$

where $k = 1,2,3,4$. The same cautionary comment made about the horizontal components applies to the vertical component of the streaming operator as well.

The final term to define in detail is the boundary term. Rather than defining an absolute boundary (*i.e.*, four explicit boundary conditions without reference to a neighboring cell), we define these conditions as though this cell is sitting in a larger array of cells. We begin by defining our cell-numbering convention (Figure 4.3).

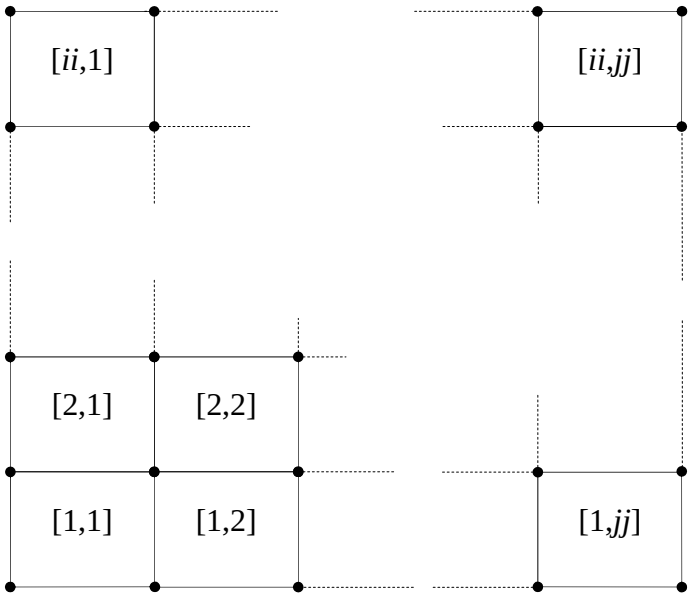

Figure 4.3 – Two-Dimensional Cell Numbering Convention

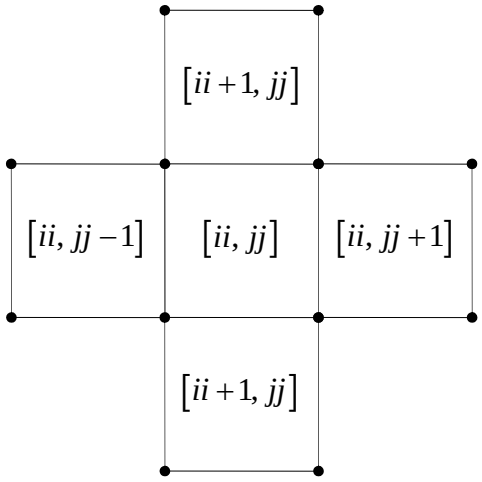

Figure 4.4 – Two-Dimensional Interior-Cell Boundary Layout

Boundary communication terms communicate the exiting flux from one cell and construct its contribution to the source term of the entrant cell. Since we have ordered our solution vector by angular quadrant, we must construct these boundary communication terms in the same way. Considering one quadrant at a time, we ask which cells and which vertices of those cells contribute to the boundary conditions? For angular quadrant one, Q_1, cell $[ii, jj-1]$, vertices 2 and 3, and cell $[ii-1, jj]$, vertices 3 and 4 contribute to the boundary condition of cell $[ii, jj]$, vertices 1, 2 and 4 (Figures 4.1 and 4.4). This results in Eq. (4.44):

$$\partial\underline{\Psi}^{(4N_1)}_{K_{(ii,jj)}} = \Theta^{(4N_1\times 4N_1)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_1)}_{K_{(ii,jj-1)}} + \mathrm{B}^{(4N_1\times 4N_1)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_1)}_{K_{(ii-1,jj)}} . \tag{4.44}$$

For angular quadrant two, Q_2, cell $[ii, jj+1]$, vertices 1 and 4, and cell $[ii-1, jj]$, vertices 3 and 4 contribute to the boundary condition of cell $[ii, jj]$, vertices 1, 2 and 3. This results in Eq. (4.45):

$$\partial\underline{\Psi}^{(4N_2)}_{K_{(ii,jj)}} = \Gamma^{(4N_2\times 4N_2)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_2)}_{K_{(ii,jj+1)}} + \mathrm{B}^{(4N_2\times 4N_2)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_2)}_{K_{(ii-1,jj)}} . \tag{4.45}$$

For angular quadrant three, Q_3, cell $[ii, jj+1]$, vertices 1 and 4, and cell $[ii+1, jj]$, vertices 1 and 2 contribute to the boundary condition of cell $[ii, jj]$, vertices 2, 3 and 4. This results in Eq. (4.46):

$$\partial\underline{\Psi}^{(4N_3)}_{K_{(ii,jj)}} = \Gamma^{(4N_3\times 4N_3)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_3)}_{K_{(ii,jj+1)}} + \mathrm{A}^{(4N_3\times 4N_3)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_3)}_{K_{(ii+1,jj)}} \ . \tag{4.46}$$

For angular quadrant four, Q_4, cell $[ii, jj-1]$, vertices 2 and 3, and cell $[ii+1, jj]$, vertices 1 and 2 contribute to the boundary condition of cell $[ii, jj]$, vertices 1, 3 and 4. This results in Eq. (4.47):

$$\partial\underline{\Psi}^{(4N_4)}_{K_{(ii,jj)}} = \Theta^{(4N_4\times 4N_4)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_4)}_{K_{(ii,jj-1)}} + \mathrm{A}^{(4N_4\times 4N_4)}_{K_{(ii,jj)}}\underline{\Psi}^{(4N_4)}_{K_{(ii+1,jj)}} \ . \tag{4.47}$$

Equations (4.44) - (4.47) would look slightly different if we did not make the choice of representing directional cosines as only positive values and not allowing them to change sign with each quadrant. Making this choice leads to nothing but positive signs on the right-hand-side of the transport equation when it is in block-matrix form. This makes physical sense since they should be adding to the magnitude of the solution. Without making this choice, negative signs would appear on the right-hand-side to compensate for the negative values of some of the directional cosines. This is not a major complication; however, the reader should be aware of this subtlety to avoid confusion, not to mention gross mathematical errors.

Equations (4.44) - (4.47) can be further abstracted to describe the entire boundary of cell $[ii, jj]$. The complete boundary term for a single cell is given by Eq. (4.48):

$$\partial\underline{\Psi}^{\ell}_{K_{(ii,jj)}} = \Theta^{\ell}_{K_{(ii,jj)}}\underline{\Psi}^{\ell}_{K_{(ii,jj-1)}} + \Gamma^{\ell}_{K_{(ii,jj)}}\underline{\Psi}^{\ell}_{K_{(ii,jj+1)}} + \mathrm{A}^{\ell}_{K_{(ii,jj)}}\underline{\Psi}^{\ell}_{K_{(ii+1,jj)}} + \mathrm{B}^{\ell}_{K_{(ii,jj)}}\underline{\Psi}^{\ell}_{\left(K_{ii-1,jj}\right)}. \tag{4.48}$$

The finer structure of the boundary communication operators in the previous expression follows from Figures 4.1 and 4.4. The left boundary communication operator of cell $[ii, jj]$ at grid level ℓ is given by Eq. (4.49):

$$\Theta^{\ell}_{K_{(ii,jj)}} = \left(\frac{h^{y}_{K_{(ii,jj)}}}{2} \right) \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \Theta^{(4N_2 \times 4N_2)}_{K_{(ii,jj)}} & 0 & 0 \\ 0 & 0 & \Theta^{(4N_3 \times 4N_3)}_{K_{(ii,jj)}} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.49}$$

where each of the two block-diagonal terms is given by Eq. (4.50):

$$\Theta^{(4N_i \times 4N_i)}_{K_{(ii,jj)}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \left(\frac{\theta^{y}_{K_{(ii,jj)}} + 1}{2\theta^{y}_{K_{(ii,jj)}}} \right) \mathrm{M}^{(N_i \times N_i)} & 0 & 0 & \left(\frac{\theta^{y}_{K_{(ii,jj)}} - 1}{2\theta^{y}_{K_{(ii,jj)}}} \right) \mathrm{M}^{(N_i \times N_i)} \\ \left(\frac{\theta^{y}_{K_{(ii,jj)}} - 1}{2\theta^{y}_{K_{(ii,jj)}}} \right) \mathrm{M}^{(N_i \times N_i)} & 0 & 0 & \left(\frac{\theta^{y}_{K_{(ii,jj)}} + 1}{2\theta^{y}_{K_{(ii,jj)}}} \right) \mathrm{M}^{(N_i \times N_i)} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.50}$$

for $i = 2,3$. Note how this notation follows the description of the right cell's contribution in Eqs. (4.45) and (4.46) – angular quadrants 2 and 3 (Eq. (4.49)) and vertices 1 and 4 (Eq. (4.50)). Entry $(2,1)$ of Eq. (4.50) represents the contribution of vertex 1 (*i.e.*, column one) of cell $[ii, jj+1]$ to the source for vertex 2 (*i.e.*, row two) of cell $[ii, jj]$. Similar interpretations greatly assist in understanding the intricate block-matrix structure we are presenting in this section. The right boundary communication operator of cell $[ii, jj]$ at grid level ℓ is given by Eq. (4.51):

$$\Gamma^{\ell}_{K_{(ii,jj)}} = \left(\frac{h^{y}_{K_{(ii,jj)}}}{2} \right) \begin{bmatrix} \Gamma^{(4N_1 \times 4N_1)}_{K_{(ii,jj)}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Gamma^{(4N_4 \times 4N_4)}_{K_{(ii,jj)}} \end{bmatrix}, \tag{4.51}$$

where the individual blocks of the previous expression are given by Eq. (4.52):

$$\Gamma_{K_{(ii,jj)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} 0 & \left(\dfrac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right)\mathrm{M}^{(N_i\times N_i)} & \left(\dfrac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right)\mathrm{M}^{(N_i\times N_i)} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \left(\dfrac{\theta^{y}_{K_{(ii,jj)}}-1}{2\theta^{y}_{K_{(ii,jj)}}}\right)\mathrm{M}^{(N_i\times N_i)} & \left(\dfrac{\theta^{y}_{K_{(ii,jj)}}+1}{2\theta^{y}_{K_{(ii,jj)}}}\right)\mathrm{M}^{(N_i\times N_i)} & 0 \end{bmatrix}, \tag{4.52}$$

for $i=1,4$. The top boundary communication operator of cell $[ii,jj]$ at grid level ℓ is given by Eq. (4.53):

$$\mathrm{A}_{K_{(ii,jj)}}^{\ell}=\left(\frac{h^{x}_{K_{(ii,jj)}}}{2}\right)\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \mathrm{A}_{K_{(ii,jj)}}^{(4N_3\times 4N_3)} & 0 \\ 0 & 0 & 0 & \mathrm{A}_{K_{(ii,jj)}}^{(4N_4\times 4N_4)} \end{bmatrix}, \tag{4.53}$$

where the individual block-diagonal terms of the previous expression are given by Eq. (4.54):

$$\mathrm{A}_{K_{(ii,jj)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \left(\dfrac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right)\mathrm{N}^{(N_i\times N_i)} & \left(\dfrac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right)\mathrm{N}^{(N_i\times N_i)} & 0 & 0 \\ \left(\dfrac{\theta^{x}_{K_{(ii,jj)}}+1}{2\theta^{x}_{K_{(ii,jj)}}}\right)\mathrm{N}^{(N_i\times N_i)} & \left(\dfrac{\theta^{x}_{K_{(ii,jj)}}-1}{2\theta^{x}_{K_{(ii,jj)}}}\right)\mathrm{N}^{(N_i\times N_i)} & 0 & 0 \end{bmatrix}, \tag{4.54}$$

for $i=3,4$. The bottom boundary communication operator of cell $[ii,jj]$ at grid level ℓ is given by Eq. (4.55):

$$\mathrm{B}^{\ell}_{K_{(ii,jj)}} = \left(\frac{h^{x}_{K_{(ii,jj)}}}{2} \right) \begin{bmatrix} \mathrm{B}^{(4N_1 \times 4N_1)}_{K_{(ii,jj)}} & 0 & 0 & 0 \\ 0 & \mathrm{B}^{(4N_2 \times 4N_2)}_{K_{(ii,jj)}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.55}$$

where each of the block-diagonal terms of the previous expression are given by Eq. (4.56):

$$\mathrm{B}^{(4N_i \times 4N_i)}_{K_{(ii,jj)}} = \begin{bmatrix} 0 & 0 & \left(\frac{\theta^{x}_{K_{(ii,jj)}} - 1}{2\theta^{x}_{K_{(ii,jj)}}} \right) \mathrm{N}^{(N_i \times N_i)} & \left(\frac{\theta^{x}_{K_{(ii,jj)}} + 1}{2\theta^{x}_{K_{(ii,jj)}}} \right) \mathrm{N}^{(N_i \times N_i)} \\ 0 & 0 & \left(\frac{\theta^{x}_{K_{(ii,jj)}} + 1}{2\theta^{x}_{K_{(ii,jj)}}} \right) \mathrm{N}^{(N_i \times N_i)} & \left(\frac{\theta^{x}_{K_{(ii,jj)}} - 1}{2\theta^{x}_{K_{(ii,jj)}}} \right) \mathrm{N}^{(N_i \times N_i)} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.56}$$

for $i = 3,4$.

With the boundary communication operators defined in Eqs. (4.49) - (4.56), we can write the system of equations that describe a BLDFEM representation of the transport equation using an arbitrary, asymmetric discrete-ordinate quadrature set for a single cell in block-matrix form:

$$\begin{aligned} &\left[\mathrm{M}^{\ell}_{K_{(ii,jj)}} + \mathrm{N}^{\ell}_{K_{(ii,jj)}} + \mathrm{T}^{\ell}_{K_{(ii,jj)}} - R^{\ell}_{K_{(ii,jj)}} \right] \underline{\Psi}^{\ell}_{K_{(ii,jj)}} = \\ &\Theta^{\ell}_{K_{(ii,jj)}} \underline{\Psi}^{\ell}_{K_{(ii,jj-1)}} + \Gamma^{\ell}_{K_{(ii,jj)}} \underline{\Psi}^{\ell}_{K_{(ii,jj+1)}} + \mathrm{A}^{\ell}_{K_{(ii,jj)}} \underline{\Psi}^{\ell}_{K_{(ii+1,jj)}} + \mathrm{B}^{\ell}_{K_{(ii,jj)}} \underline{\Psi}^{\ell}_{K_{(ii-1,jj)}} + \underline{Q}^{\ell}_{K_{(ii,jj)}}. \end{aligned} \tag{4.57}$$

From this point forward we will be discussing the angular flux correction; thus, we rewrite Eq. (4.57) to reflect this with Eq. (4.58):

$$\begin{aligned} &\left[\mathrm{M}^{\ell}_{K_{(ii,jj)}} + \mathrm{N}^{\ell}_{K_{(ii,jj)}} + \mathrm{T}^{\ell}_{K_{(ii,jj)}} - R^{\ell}_{K_{(ii,jj)}} \right] \underline{\psi}^{\ell}_{K_{(ii,jj)}} = \\ &\Theta^{\ell}_{K_{(ii,jj)}} \underline{\psi}^{\ell}_{K_{(ii,jj-1)}} + \Gamma^{\ell}_{K_{(ii,jj)}} \underline{\psi}^{\ell}_{K_{(ii,jj+1)}} + \mathrm{A}^{\ell}_{K_{(ii,jj)}} \underline{\psi}^{\ell}_{K_{(ii+1,jj)}} + \mathrm{B}^{\ell}_{K_{(ii,jj)}} \underline{\psi}^{\ell}_{K_{(ii-1,jj)}}. \end{aligned} \tag{4.58}$$

Four-Cell Operator Inversion

A logical extension of the one-dimensional grid-coarsening procedure presented in Chapter II (*i.e.*, two fine-grid cells merged to form a single, coarser-grid cell) is coarsening a two-by-two *quad* of cells into a single coarser-grid cell. The goals of this section and Section IV are: 1) use the block-matrix structure derived in the previous section to form a four-cell transport operator; 2) decompose this four-cell operator in the manner of Sherman-Morrison for efficient inversion; 3) discuss the spatial shape of the error following relaxation using the four-cell inverse; and, 4) introduce the notion of the slowest-converging error mode. The essential difference between this discussion and the one-dimensional equivalent lies in the error characterization following relaxation. An analogous detailed algebraic discussion is intractable, so instead we use Fourier analysis to characterize the slowest-converging error mode.

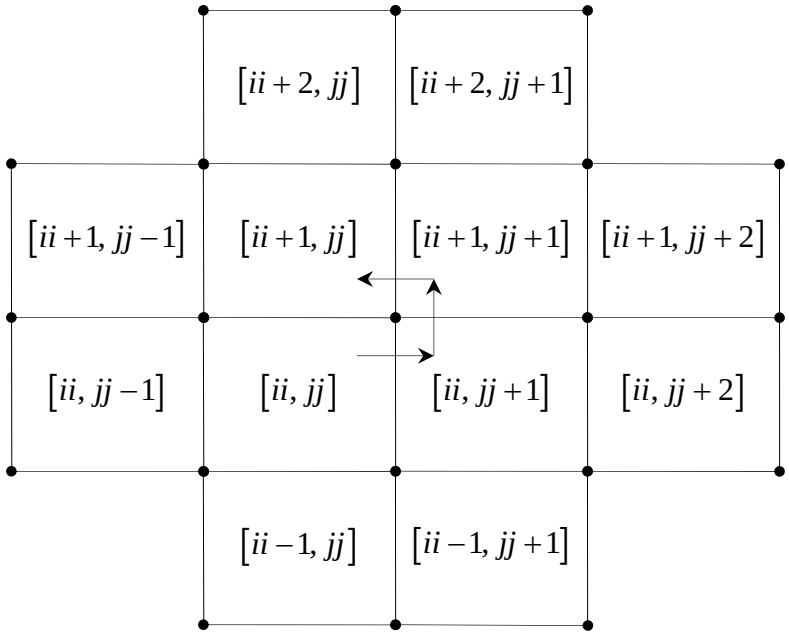

Figure 4.5 – Visualization of an Interior Quad and Block-Matrix Order

Using Eq. (4.58) and the four-cell block-matrix ordering described by Figure 4.5, it is simple to construct a four-cell transport operator. Using the cell-numbering scheme in Figure 4.3, we construct Eq. (4.59) – a local four-cell problem using only intra-quad boundary communication operators:

$$\begin{bmatrix} \Pi^{\ell}_{K_{(ii,jj)}} & -\Theta^{\ell}_{K_{(ii,jj)}} & 0 & -\mathrm{A}^{\ell}_{K_{(ii,jj)}} \\ -\Gamma^{\ell}_{K_{(ii,jj+1)}} & \Pi^{\ell}_{K_{(ii,jj+1)}} & -\mathrm{A}^{\ell}_{K_{(ii,jj+1)}} & 0 \\ 0 & -\mathrm{B}^{\ell}_{K_{(ii+1,jj+1)}} & \Pi^{\ell}_{K_{(ii+1,jj+1)}} & -\Gamma^{\ell}_{K_{(ii+1,jj+1)}} \\ -\mathrm{B}^{\ell}_{K_{(ii,jj+1)}} & 0 & -\Theta^{\ell}_{K_{(ii,jj+1)}} & \Pi^{\ell}_{K_{(ii,jj+1)}} \end{bmatrix} \begin{bmatrix} \underline{\Psi}^{\ell}_{K_{(ii,jj)}} \\ \underline{\Psi}^{\ell}_{K_{(ii,jj+1)}} \\ \underline{\Psi}^{\ell}_{K_{(ii+1,jj+1)}} \\ \underline{\Psi}^{\ell}_{K_{(ii,jj+1)}} \end{bmatrix} = \begin{bmatrix} \partial\underline{\Psi}^{\ell}_{K_{(ii,jj)}} \\ \partial\underline{\Psi}^{\ell}_{K_{(ii,jj+1)}} \\ \partial\underline{\Psi}^{\ell}_{K_{(ii+1,jj+1)}} \\ \partial\underline{\Psi}^{\ell}_{K_{(ii,jj+1)}} \end{bmatrix}. \tag{4.59}$$

The right-hand-side of Eq. (4.59) represents the boundary of the entire four-cell quad, and we have defined the block-diagonal terms with Eq. (4.60):

$$\Pi^{\ell}_{K_{(ii,jj)}} = \left[\mathrm{M}^{\ell}_{K_{(ii,jj)}} + \mathrm{N}^{\ell}_{K_{(ii,jj)}} + \mathrm{T}^{\ell}_{K_{(ii,jj)}} - R^{\ell}_{K_{(ii,jj)}} \right]. \tag{4.60}$$

For the general solver, we do not consider the cell ordering for the block-matrix describing the entire problem; however, the complete cell ordering will have to be considered for the Fourier analysis – this is discussed in the next chapter.

With the four-cell operator defined in Eq. (4.59), we can decompose it in terms of the Sherman-Morrison formula for the inverse. Recall that Sherman-Morrison is simply a guide for forming the inverse. Restricting the problem to isotropic scattering limits the largest full matrix requiring inversion to 16×16 regardless of the quadrature order. The decomposition begins by writing the four-cell operator in the manner of Eq. (4.61):

$$\begin{bmatrix} \Pi^{\ell}_{K_{(ii,jj)}} & -\Theta^{\ell}_{K_{(ii,jj)}} & 0 & -\mathrm{A}^{\ell}_{K_{(ii,jj)}} \\ -\Gamma^{\ell}_{K_{(ii,jj+1)}} & \Pi^{\ell}_{K_{(ii,jj+1)}} & -\mathrm{A}^{\ell}_{K_{(ii,jj+1)}} & 0 \\ 0 & -\mathrm{B}^{\ell}_{K_{(ii+1,jj+1)}} & \Pi^{\ell}_{K_{(ii+1,jj+1)}} & -\Gamma^{\ell}_{K_{(ii+1,jj+1)}} \\ -\mathrm{B}^{\ell}_{K_{(ii,jj+1)}} & 0 & -\Theta^{\ell}_{K_{(ii+1,jj)}} & \Pi^{\ell}_{K_{(ii+1,jj)}} \end{bmatrix} = \Pi^{\ell}_{0} - VW^{T}, \tag{4.61}$$

where Π_0^{ℓ} corresponds to all streaming-rate-density, collision-rate-density and intra-quad boundary-communication information; and, VW^T contains all scattering-rate-density information and is also to simplest to detail. For Sherman-Morrison to be efficient, both V and W^T should be of less-than-full rank. This is certainly true of W^T, which simply lists the quadrature weights:

$$\underline{W}^T = \begin{bmatrix} W^T & 0 & 0 & 0 \\ 0 & W^T & 0 & 0 \\ 0 & 0 & W^T & 0 \\ 0 & 0 & 0 & W^T \end{bmatrix}, \tag{4.62}$$

where the block-diagonal terms of Eq. (4.62):

$$W^T = \begin{bmatrix} W^{(4\times N_1)} & W^{(4\times N_2)} & W^{(4\times N_3)} & W^{(4\times N_4)} \end{bmatrix}; \tag{4.63}$$

and, the individual blocks in Eq. (4.63) are defined in Eq. (4.24). The dimensions of the matrix defined in Eq. (4.62) are $\left[16\times 4\left(N_1+N_2+N_3+N_4\right)\right]$, and it is of rank sixteen (16). The matrix V is defined in Eq. (4.64) and reflects the scaling of the scalar flux into a number of repeated entries in the column vector, and the effect of multiply by the finite element mass matrix:

$$V = \begin{bmatrix} V_{K_{(ii,jj)}} & 0 & 0 & 0 \\ 0 & V_{K_{(ii,jj+1)}} & 0 & 0 \\ 0 & 0 & V_{K_{(ii+1,jj+1)}} & 0 \\ 0 & 0 & 0 & V_{K_{(ii+1,jj)}} \end{bmatrix}, \tag{4.64}$$

where the individual block-diagonal terms of the previous expression are given by Eqs. (4.65) and (4.66):

$$V_{K_{(ii,jj)}} = \begin{bmatrix} V_{K_{(ii,jj)}}^{(4N_1\times 4)} \\ V_{K_{(ii,jj)}}^{(4N_2\times 4)} \\ V_{K_{(ii,jj)}}^{(4N_3\times 4)} \\ V_{K_{(ii,jj)}}^{(4N_4\times 4)} \end{bmatrix}, \tag{4.65}$$

$$V_{K_{(ii,jj)}}^{(4N_i\times 4)} =$$

$$\begin{bmatrix} \left(\theta_{xy}^{++}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{-+}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{--}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{+-}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} \\ \left(\theta_{xy}^{-+}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{++}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{+-}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{--}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} \\ \left(\theta_{xy}^{--}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{+-}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{++}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{-+}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} \\ \left(\theta_{xy}^{+-}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{--}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{-+}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} & \left(\theta_{xy}^{++}\right)_{K_{(ii,jj)}} \underline{1}^{(N_i)} \end{bmatrix}. \tag{4.66}$$

Recall that the coefficients in Eq. (4.66) are defined in Eqs. (4.12) - (4.15). The matrix defined in Eq. (4.64) is of dimension $\left[4\left(N_1+N_2+N_3+N_4\right)\times 16\right]$, and it has rank sixteen (16). With these definitions, we have successfully decomposed the scattering operator into two matrices, both of rank sixteen (16), for a general S_N quadrature set. The next step in the four-cell inversion process is to efficiently invert the sparse matrix, Π_0^{ℓ}.

The sparse-matrix-component of the four-cell operator is angularly decoupled. This property allows us to construct the sparse-matrix inverse without inverting any matrix that is larger then 4×4. Writing the streaming and collision rate density component of Eq. (4.61), we obtain Eq. (4.67):

$$\Pi_0^{\ell} = \begin{bmatrix} \Pi_{K_{(ii,jj),0}}^{\ell} & -\Theta_{K_{(ii,jj)}}^{\ell} & 0 & -\mathrm{A}_{K_{(ii,jj)}}^{\ell} \\ -\Gamma_{K_{(ii,jj+1)}}^{\ell} & \Pi_{K_{(ii,jj+1),0}}^{\ell} & -\mathrm{A}_{K_{(ii,jj+1)}}^{\ell} & 0 \\ 0 & -\mathrm{B}_{K_{(ii+1,jj+1)}}^{\ell} & \Pi_{K_{(ii+1,jj+1),0}}^{\ell} & -\Gamma_{K_{(ii+1,jj+1)}}^{\ell} \\ -\mathrm{B}_{K_{(ii,jj+1)}}^{\ell} & 0 & -\Theta_{K_{(ii+1,jj)}}^{\ell} & \Pi_{K_{(ii+1,jj),0}}^{\ell} \end{bmatrix}, \tag{4.67}$$

where the block-diagonal terms of the previous expression are given by Eq. (4.68):

$$\Pi^{\ell}_{K_{(ii,jj),0}} = \begin{bmatrix} \Pi^{(4N_1\times 4N_1)}_{K_{(ii,jj)}} & 0 & 0 & 0 \\ 0 & \Pi^{(4N_2\times 4N_2)}_{K_{(ii,jj)}} & 0 & 0 \\ 0 & 0 & \Pi^{(4N_3\times 4N_3)}_{K_{(ii,jj)}} & 0 \\ 0 & 0 & 0 & \Pi^{(4N_4\times 4N_4)}_{K_{(ii,jj)}} \end{bmatrix}. \tag{4.68}$$

The off-block-diagonal entries are defined in Eqs. (4.49) - (4.56); however, for this derivation, we give a modified presentation of their block-matrix form for clarity in Eqs. (4.69) - (4.76):

$$\Theta^{\ell}_{K_{(ii,jj)}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \Pi^{(4N_2\times 4N_2)}_{K_{(ii,jj+1)}\to K_{(ii,jj)}} & 0 & 0 \\ 0 & 0 & \Pi^{(4N_3\times 4N_3)}_{K_{(ii,jj+1)}\to K_{(ii,jj)}} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.69}$$

$$\Theta^{\ell}_{K_{(ii+1,jj)}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \Pi^{(4N_2\times 4N_2)}_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}} & 0 & 0 \\ 0 & 0 & \Pi^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.70}$$

$$\mathrm{A}^{\ell}_{K_{(ii,jj)}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \Pi^{(4N_3\times 4N_3)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} & 0 \\ 0 & 0 & 0 & \Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} \end{bmatrix}, \tag{4.71}$$

$$\mathrm{A}^{\ell}_{K_{(ii,jj+1)}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \Pi^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} & 0 \\ 0 & 0 & 0 & \Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} \end{bmatrix}, \tag{4.72}$$

$$\Gamma^{\ell}_{K_{(ii,jj+1)}} = \begin{bmatrix} \Pi^{(4N_1\times 4N_1)}_{K_{(ii,jj)}\to K_{(ii,jj+1)}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Pi^{(4N_4\times 4N_4)}_{K_{(ii,jj)}\to K_{(ii,jj+1)}} \end{bmatrix}, \tag{4.73}$$

$$\Gamma^{\ell}_{K_{(ii+1,jj+1)}} = \begin{bmatrix} \Pi^{(4N_1\times 4N_1)}_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}} \end{bmatrix}, \tag{4.74}$$

$$\mathrm{B}^{\ell}_{K_{(ii+1,jj+1)}} = \begin{bmatrix} \Pi^{(4N_1\times 4N_1)}_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}} & 0 & 0 & 0 \\ 0 & \Pi^{(4N_2\times 4N_2)}_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4.75}$$

$$\mathrm{B}^{\ell}_{K_{(ii+1,jj)}} = \begin{bmatrix} \Pi^{(4N_1\times 4N_1)}_{K_{(ii,jj)}\to K_{(ii+1,jj)}} & 0 & 0 & 0 \\ 0 & \Pi^{(4N_2\times 4N_2)}_{K_{(ii,jj)}\to K_{(ii+1,jj)}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{4.76}$$

Each non-zero block in Eqs. (4.68) - (4.76) has a distinct physical interpretation. The diagonal blocks of Eq. (4.68) represent the within-cell transport of particles in cell $K_{(ii,jj)}$ along the discrete-ordinates in each of the four angular quadrants. Recall there is no scattering represented in the sparse matrix, thus there are no operators coupling one

angular quadrant to another. The non-zero blocks in Eqs. (4.69) - (4.76) represent the boundary communication terms. Note that there are no blocks representing communication between two cells that join at a single point as opposed to a surface. This makes physical sense; however, the inverse of the complete sparse matrix will describe all cell-to-cell communications. The sparse matrix inverse can be written as Eq. (4.77):

$$\left(\Pi_0^{\ell}\right)^{-1} = \begin{bmatrix} \mathrm{Z}^{\ell}_{K_{(ii,jj)}} & \mathrm{Z}^{\ell}_{K_{(ii,jj+1)}\to K_{(ii,jj)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj+1)}\to K_{(ii,jj)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} \\ \mathrm{Z}^{\ell}_{K_{(ii,jj)}\to K_{(ii,jj+1)}} & \mathrm{Z}^{\ell}_{K_{(ii,jj+1)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj)}\to K_{(ii,jj+1)}} \\ \mathrm{Z}^{\ell}_{K_{(ii,jj)}\to K_{(ii+1,jj+1)}} & \mathrm{Z}^{\ell}_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj+1)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}} \\ \mathrm{Z}^{\ell}_{K_{(ii,jj)}\to K_{(ii,jj+1)}} & \mathrm{Z}^{\ell}_{K_{(ii,jj+1)}\to K_{(ii+1,jj)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}} & \mathrm{Z}^{\ell}_{K_{(ii+1,jj)}} \end{bmatrix}. \tag{4.77}$$

To most efficiently construct the inverse of Eq. (4.77), attention must be paid to the order of operations. The block-diagonal entries must be computed first. They are given by Eq. (4.78):

$$\mathrm{Z}^{\ell}_{K_{(ii,jj)}} = \begin{bmatrix} \mathrm{Z}^{(4N_1\times 4N_1)}_{K_{(ii,jj)}} & 0 & 0 & 0 \\ 0 & \mathrm{Z}^{(4N_2\times 4N_2)}_{K_{(ii,jj)}} & 0 & 0 \\ 0 & 0 & \mathrm{Z}^{(4N_3\times 4N_3)}_{K_{(ii,jj)}} & 0 \\ 0 & 0 & 0 & \mathrm{Z}^{(4N_4\times 4N_4)}_{K_{(ii,jj)}} \end{bmatrix}, \tag{4.78}$$

for all $K_{(ii,jj)}$, where each diagonal block of Eq. (4.78) is given by Eq. (4.79):

$$\mathrm{Z}^{(4N_i\times 4N_i)}_{K_{(ii,jj)}} = \left[\Pi^{(4N_i\times 4N_i)}_{K_{(ii,jj)}}\right]^{-1}, \tag{4.79}$$

for $i = 1,2,3,4$. Recall that each block in Eq. (4.79) is a diagonal matrix of dimension $\left(N_i \times N_i\right)$; thus, the complete inverse can be formed by inverting a series of $\left(4\times 4\right)$ matrices that are full, in general. Following the formation of each inverse along the block-diagonal, the blocks representing cell-to-cell communication for cells share a surface can be inverted. Following this procedure, the block representing cell-to-cell

communication between cells that have no common surface, but do have a common vertex, can be inverted. We implemented this procedure by nesting tasks according to angular quadrant. For each quadrant, the block-inverse are formed as follows:

Q_1, within-cell operators are given by Eqs. (4.80) - (4.83):

$$Z_{K_{(ii,jj)}}^{(4N_1\times 4N_1)} = \left[\Pi_{K_{(ii,jj)}}^{(4N_1\times 4N_1)}\right]^{-1}, \tag{4.80}$$

$$Z_{K_{(ii,jj+1)}}^{(4N_1\times 4N_1)} = \left[\Pi_{K_{(ii,jj+1)}}^{(4N_1\times 4N_1)}\right]^{-1}, \tag{4.81}$$

$$Z_{K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} = \left[\Pi_{K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)}\right]^{-1}, \tag{4.82}$$

$$Z_{K_{(ii+1,jj)}}^{(4N_1\times 4N_1)} = \left[\Pi_{K_{(ii+1,jj)}}^{(4N_1\times 4N_1)}\right]^{-1}; \tag{4.83}$$

Q_1, cell-to-cell-through-surface operators are given by Eqs. (4.84) - (4.87):

$$Z_{K_{(ii,jj)}\to K_{(ii,jj+1)}}^{(4N_1\times 4N_1)} = Z_{K_{(ii,jj+1)}}^{(4N_1\times 4N_1)}\left[\Pi_{K_{(ii,jj)}\to K_{(ii,jj+1)}}^{(4N_1\times 4N_1)} Z_{K_{(ii,jj)}}^{(4N_1\times 4N_1)}\right], \tag{4.84}$$

$$Z_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} = Z_{K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)}\left[\Pi_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} Z_{K_{(ii,jj+1)}}^{(4N_1\times 4N_1)}\right], \tag{4.85}$$

$$Z_{K_{(ii,jj)}\to K_{(ii+1,jj)}}^{(4N_1\times 4N_1)} = Z_{K_{(ii+1,jj)}}^{(4N_1\times 4N_1)}\left[\Pi_{K_{(ii,jj)}\to K_{(ii+1,jj)}}^{(4N_1\times 4N_1)} Z_{K_{(ii,jj)}}^{(4N_1\times 4N_1)}\right], \tag{4.86}$$

$$Z_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} = Z_{K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)}\left[\Pi_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} Z_{K_{(ii+1,jj)}}^{(4N_1\times 4N_1)}\right]; \tag{4.87}$$

Q_1, cell-to-cell-through-other-cells operator is given by Eq. (4.88):

$$Z_{K_{(ii,jj)}\to K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} = Z_{K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)}\left[\begin{array}{l}\Pi_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} Z_{K_{(ii,jj)}\to K_{(ii,jj+1)}}^{(4N_1\times 4N_1)} + \\ \Pi_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}}^{(4N_1\times 4N_1)} Z_{K_{(ii,jj)}\to K_{(ii+1,jj)}}^{(4N_1\times 4N_1)}\end{array}\right]. \tag{4.88}$$

Q_2, within-cell operators are given by Eqs. (4.89) - (4.92):

$$Z_{K_{(ii,jj)}}^{(4N_2\times 4N_2)} = \left[\Pi_{K_{(ii,jj)}}^{(4N_2\times 4N_2)}\right]^{-1}, \tag{4.89}$$

$$Z_{K_{(ii,jj+1)}}^{(4N_2\times 4N_2)} = \left[\Pi_{K_{(ii,jj+1)}}^{(4N_2\times 4N_2)}\right]^{-1}, \tag{4.90}$$

$$Z_{K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)} = \left[\Pi_{K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)}\right]^{-1}, \tag{4.91}$$

$$Z_{K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} = \left[\Pi_{K_{(ii+1,jj)}}^{(4N_2\times 4N_2)}\right]^{-1}; \tag{4.92}$$

Q_2, cell-to-cell-through-surface operators are given by Eqs. (4.93) - (4.96):

$$Z_{K_{(ii,jj+1)}\to K_{(ii,jj)}}^{(4N_2\times 4N_2)} = Z_{K_{(ii,jj)}}^{(4N_2\times 4N_2)}\left[\Pi_{K_{(ii,jj+1)}\to K_{(ii,jj)}}^{(4N_2\times 4N_2)} Z_{K_{(ii,jj+1)}}^{(4N_2\times 4N_2)}\right], \tag{4.93}$$

$$Z_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)} = Z_{K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)}\left[\Pi_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)} Z_{K_{(ii,jj+1)}}^{(4N_2\times 4N_2)}\right], \tag{4.94}$$

$$Z_{K_{(ii,jj)}\to K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} = Z_{K_{(ii+1,jj)}}^{(4N_2\times 4N_2)}\left[\Pi_{K_{(ii,jj)}\to K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} Z_{K_{(ii,jj)}}^{(4N_2\times 4N_2)}\right], \tag{4.95}$$

$$Z_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} = Z_{K_{(ii+1,jj)}}^{(4N_2\times 4N_2)}\left[\Pi_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} Z_{K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)}\right]; \tag{4.96}$$

Q_2, cell-to-cell-through-other-cells operator is given by Eq. (4.97):

$$Z_{K_{(ii,jj+1)}\to K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} = Z_{K_{(ii+1,jj)}}^{(4N_2\times 4N_2)}\left[\begin{array}{l}\Pi_{K_{(ii,jj)}\to K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} Z_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)} + \\ \Pi_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}}^{(4N_2\times 4N_2)} Z_{K_{(ii,jj+1)}\to K_{(ii+1,jj+1)}}^{(4N_2\times 4N_2)}\end{array}\right]. \tag{4.97}$$

Q_3, within-cell operators are given by Eqs. (4.98) - (4.101):

$$Z_{K_{(ii,jj)}}^{(4N_3\times 4N_3)} = \left[\Pi_{K_{(ii,jj)}}^{(4N_3\times 4N_3)}\right]^{-1}, \tag{4.98}$$

$$Z_{K_{(ii,jj+1)}}^{(4N_3\times 4N_3)} = \left[\Pi_{K_{(ii,jj+1)}}^{(4N_3\times 4N_3)}\right]^{-1}, \tag{4.99}$$

$$Z_{K_{(ii+1,jj+1)}}^{(4N_3\times 4N_3)} = \left[\Pi_{K_{(ii+1,jj+1)}}^{(4N_3\times 4N_3)}\right]^{-1}, \tag{4.100}$$

$$Z_{K_{(ii+1,jj)}}^{(4N_3\times 4N_3)} = \left[\Pi_{K_{(ii+1,jj)}}^{(4N_3\times 4N_3)}\right]^{-1}; \tag{4.101}$$

Q_3, cell-to-cell-through-surface operators are given by Eqs. (4.102) - (4.105):

$$Z^{(4N_3\times 4N_3)}_{K_{(ii,jj+1)}\to K_{(ii,jj)}} = Z^{(4N_3\times 4N_3)}_{K_{(ii,jj)}}\left[\Pi^{(4N_3\times 4N_3)}_{K_{(ii,jj+1)}\to K_{(ii,jj)}} Z^{(4N_3\times 4N_3)}_{K_{(ii,jj+1)}}\right], \tag{4.102}$$

$$Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} = Z^{(4N_3\times 4N_3)}_{K_{(ii,jj)}}\left[\Pi^{(4N_3\times 4N_3)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj)}}\right], \tag{4.103}$$

$$Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} = Z^{(4N_3\times 4N_3)}_{K_{(ii,jj+1)}}\left[\Pi^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}}\right], \tag{4.104}$$

$$Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}} = Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj)}}\left[\Pi^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}} Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}}\right]; \tag{4.105}$$

Q_3, cell-to-cell-through-other-cells operator is given by Eq. (4.106):

$$Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj)}} = Z^{(4N_3\times 4N_3)}_{K_{(ii,jj)}}\left[\begin{array}{l}\Pi^{(4N_3\times 4N_3)}_{K_{(ii,jj+1)}\to K_{(ii,jj)}} Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} + \\ \Pi^{(4N_3\times 4N_3)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} Z^{(4N_3\times 4N_3)}_{K_{(ii+1,jj+1)}\to K_{(ii+1,jj)}}\end{array}\right]. \tag{4.106}$$

Q_4, within-cell operators are given by Eqs. (4.107) - (4.110):

$$Z^{(4N_4\times 4N_4)}_{K_{(ii,jj)}} = \left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii,jj)}}\right]^{-1}, \tag{4.107}$$

$$Z^{(4N_4\times 4N_4)}_{K_{(ii,jj+1)}} = \left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii,jj+1)}}\right]^{-1}, \tag{4.108}$$

$$Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}} = \left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}}\right]^{-1}, \tag{4.109}$$

$$Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}} = \left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}}\right]^{-1}; \tag{4.110}$$

Q_4, cell-to-cell-through-surface operators is given by Eqs. (4.111) - (4.114):

$$Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} = Z^{(4N_4\times 4N_4)}_{K_{(ii,jj)}}\left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}}\right], \tag{4.111}$$

$$Z^{(4N_4\times 4N_4)}_{K_{(ii,jj)}\to K_{(ii,jj+1)}} = Z^{(4N_4\times 4N_4)}_{K_{(ii,jj+1)}}\left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii,jj)}\to K_{(ii,jj+1)}} Z^{(4N_4\times 4N_4)}_{K_{(ii,jj)}}\right], \tag{4.112}$$

$$Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} = Z^{(4N_4\times 4N_4)}_{K_{(ii,jj+1)}}\left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}}\right], \tag{4.113}$$

$$Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}} = Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}}\left[\Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}} Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}}\right]; \tag{4.114}$$

Q_4, cell-to-cell-through-other-cells operator is given by Eq. (4.115):

$$Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii,jj+1)}} = Z^{(4N_4\times 4N_4)}_{K_{(ii,jj+1)}}\left[\begin{array}{l}\Pi^{(4N_4\times 4N_4)}_{K_{(ii,jj)}\to K_{(ii,jj+1)}} Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii,jj)}} + \\ \Pi^{(4N_4\times 4N_4)}_{K_{(ii+1,jj+1)}\to K_{(ii,jj+1)}} Z^{(4N_4\times 4N_4)}_{K_{(ii+1,jj)}\to K_{(ii+1,jj+1)}}\end{array}\right]. \tag{4.115}$$

There is no calculation in Eqs. (4.80) - (4.115) that requires the inversion of a matrix larger than (4×4) or the multiplication of two matrices larger than (4×4).

With all of the pieces defined that are required for the Sherman-Morrison formula, we can examine the expense of performing the four-cell relaxation step. The inverse is not formed; however, the pieces are formed and then used in the relaxation step:

$$\left[\begin{array}{cccc}\Pi^{\ell}_{K_{(ii,jj)}} & -\Theta^{\ell}_{K_{(ii,jj)}} & 0 & -\mathrm{A}^{\ell}_{K_{(ii,jj)}} \\ -\Gamma^{\ell}_{K_{(ii,jj+1)}} & \Pi^{\ell}_{K_{(ii,jj+1)}} & -\mathrm{A}^{\ell}_{K_{(ii,jj+1)}} & 0 \\ 0 & -\mathrm{B}^{\ell}_{K_{(ii+1,jj+1)}} & \Pi^{\ell}_{K_{(ii+1,jj+1)}} & -\Gamma^{\ell}_{K_{(ii+1,jj+1)}} \\ -\mathrm{B}^{\ell}_{K_{(ii,jj+1)}} & 0 & -\Theta^{\ell}_{K_{(ii+1,jj)}} & \Pi^{\ell}_{K_{(ii+1,jj)}}\end{array}\right]^{-1}\left[\begin{array}{c}\partial\underline{\psi}^{\ell}_{K_{(ii,jj)}} \\ \partial\underline{\psi}^{\ell}_{K_{(ii,jj+1)}} \\ \partial\underline{\psi}^{\ell}_{K_{(ii+1,jj+1)}} \\ \partial\underline{\psi}^{\ell}_{K_{(ii+1,jj)}}\end{array}\right] = \tag{4.116}$$

$$\left[\Pi_0^{-1} + \Pi_0^{-1}V\left(I - W^T\Pi_0^{-1}V\right)^{-1}W^T\Pi_0^{-1}\right]\partial\underline{\psi}^{\ell}.$$

The largest computation performed in Eq. (4.116) is the operation of the sparse matrix, Π_0^{-1}, on the incident-flux vector, $\partial\underline{\psi}^{\ell}$. This operation requires $\left[16\left(N_1+N_2+N_3+N_4\right)\times 16\left(N_1+N_2+N_3+N_4\right)\right]\times\left[16\left(N_1+N_2+N_3+N_4\right)\times 1\right]$ steps.

The preceding discussion shows how the four-cell inverse can be efficiently computed, and how this is used to perform four-cell relaxation. Following relaxation, we want to characterize the shape of the error on each four-cell quad. This type of characterization in one-dimension was accomplished through a detailed algebraic study. The equivalent algebraic study in two dimensions, while possible, is virtually intractable. The four-cell, S_2 (*i.e.*, the simplest) case requires the simultaneous solution of sixty-four

(64) equations and sixty-four (64) unknowns. Such a large set of equations is best approached numerically.

Error Characterization Following Relaxation

To characterize the shape of the error following relaxation, we numerically examine cases representing a range of physical problems. In the one-dimensional case, we form an unit incident isotropic flux vector and operate on it with the two-cell inverse. As a starting point, we form an analogous boundary vector and operate on it with the four-cell inverse defined in Eq. (4.116). For these characterizations, we use a set of nine physical problems representing a range of grid and material properties. Table 4.1 describes the naming convention for these nine test problems:

Table 4.1 – Naming Convention for Relaxation Test Suite Problems

$\sigma_t h_x = \sigma_t h_x$	$c = 0.999999$	$c = 0.99$	$c = 0.5$
1.0×10^{-2}	Prob_0.01_1	Prob_0.01_2	Prob_0.01_3
1.0×10^{0}	Prob_1.0_1	Prob_1.0_2	Prob_1.0_3
1.0×10^{2}	Prob_100.0_1	Prob_100.0_2	Prob_100.0_3

The Fourier analysis of the two-dimensional multigrid iteration matrix is presented in the next chapter; however, we present results from it to further the current discussion. At a given grid level, we want to characterize the slowest converging error mode. This characterization will then define the kink-factors that will be part of the prolongation operator that builds the coarse-grid operators. The resulting operator has thus been designed to eliminate the slowest converging error mode that we characterized on the previous grid level.

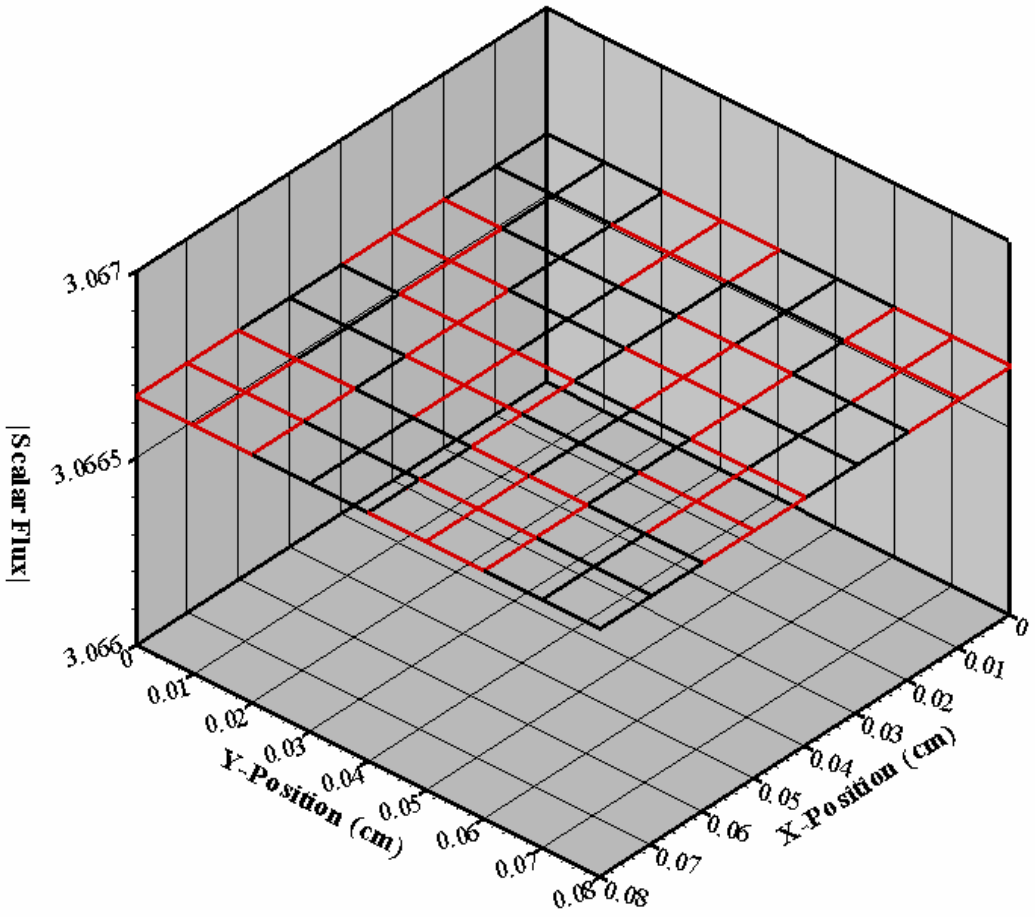

Figure 4.6 – Slowest Converging Error Mode for Problem_0.01_1

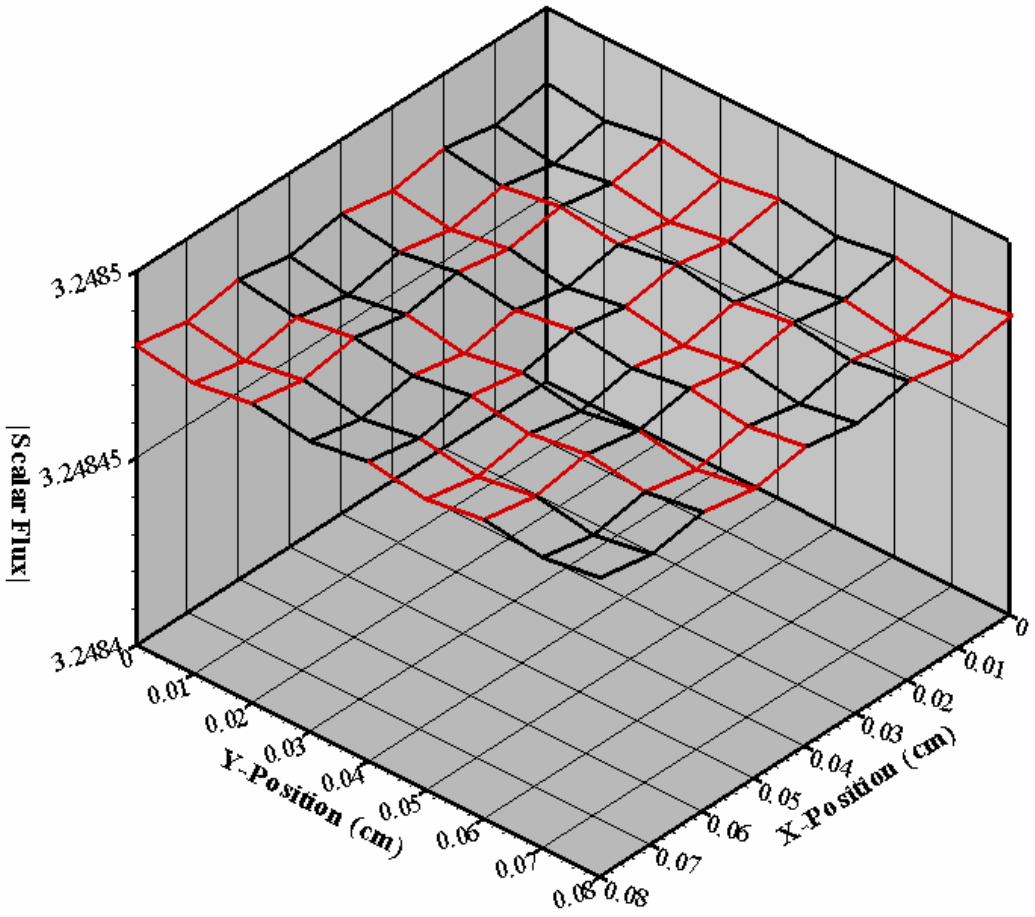

Figure 4.7 – Slowest Converging Error mode for Problem_0.01_2

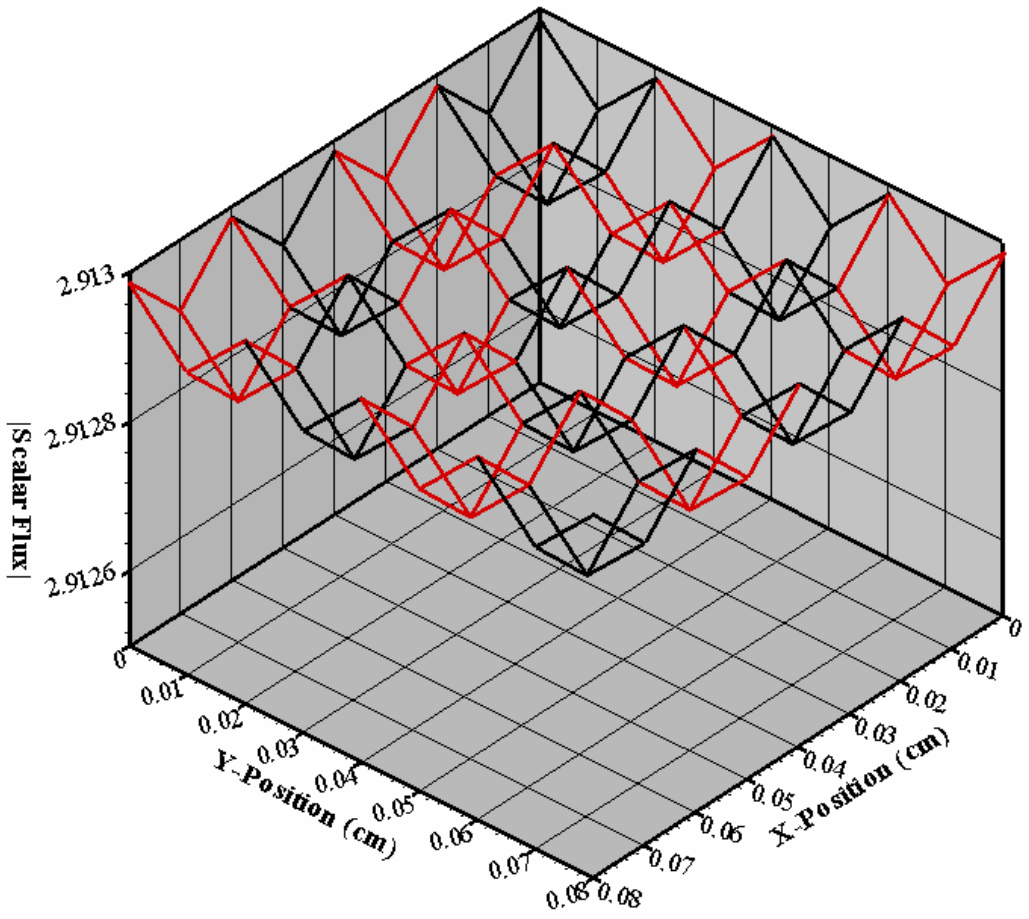

Figure 4.8 – Slowest Converging Error mode for Problem_0.01_3

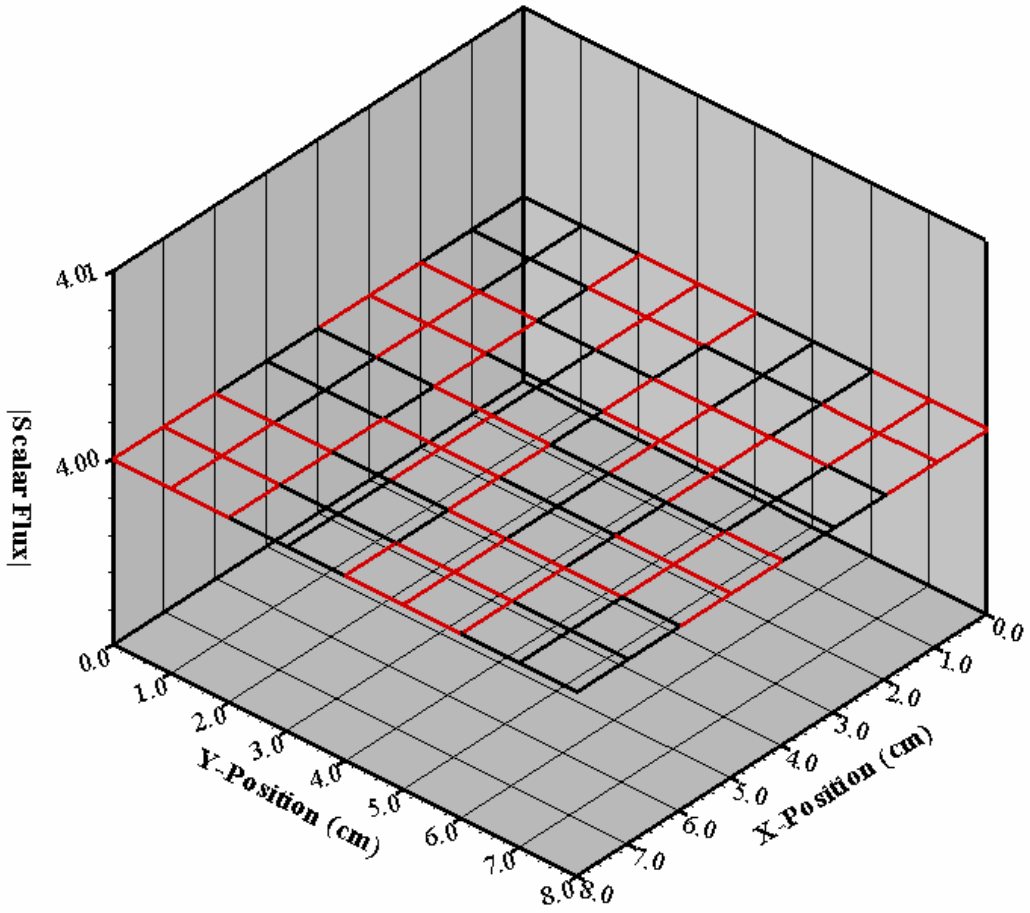

Figure 4.9 – Slowest Converging Error mode for Problem_1.0_1

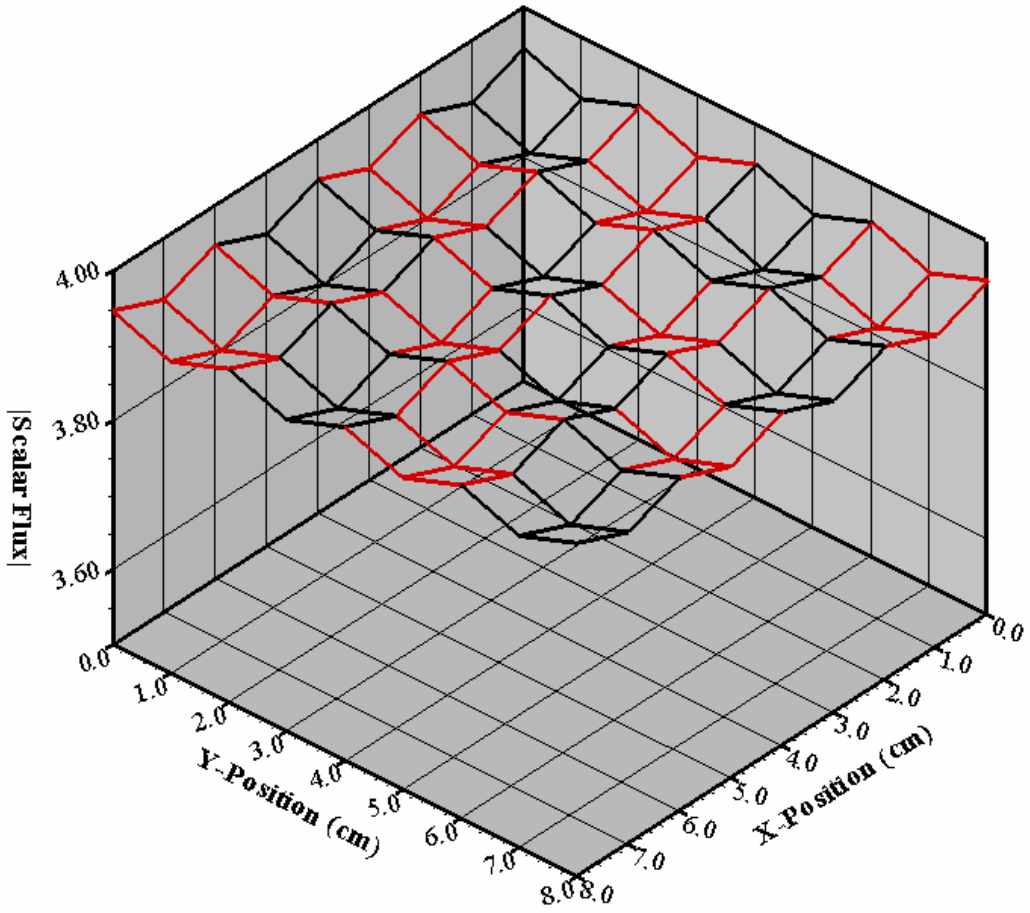

Figure 4.10 – Slowest Converging Error mode for Problem_1.0_2

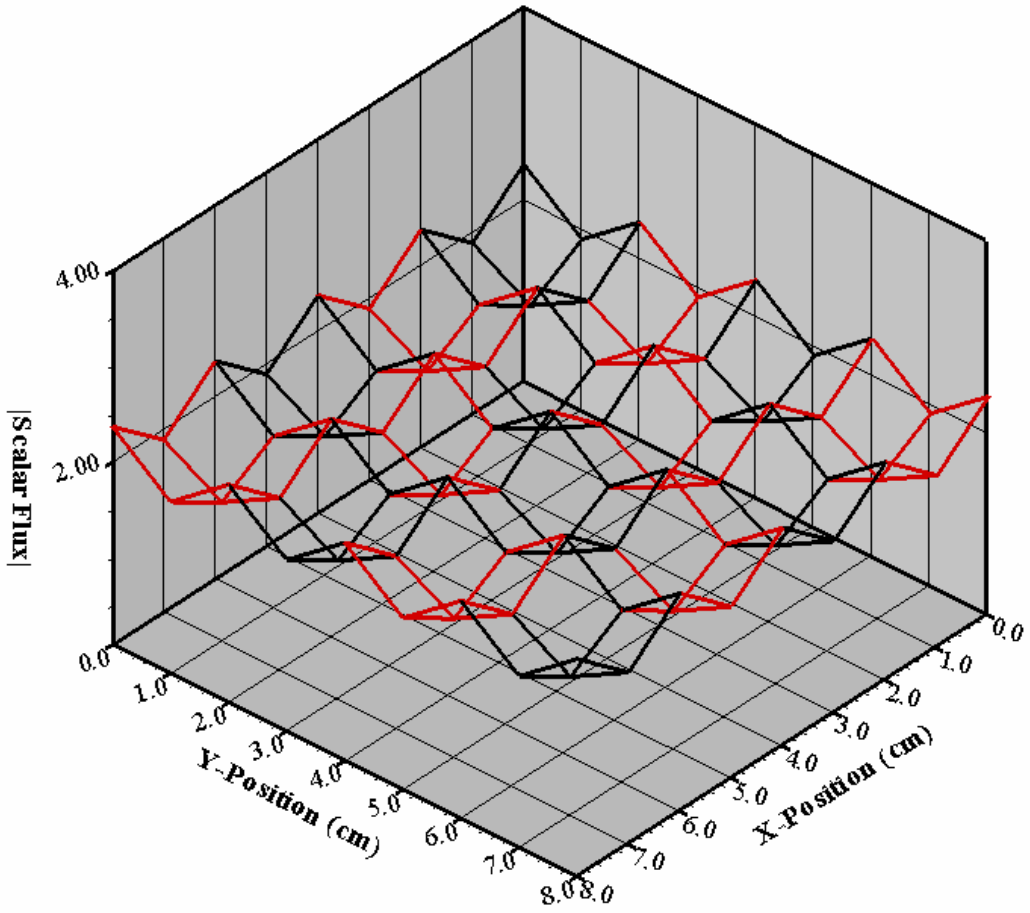

Figure 4.11 – Slowest Converging Error mode for Problem_1.0_3

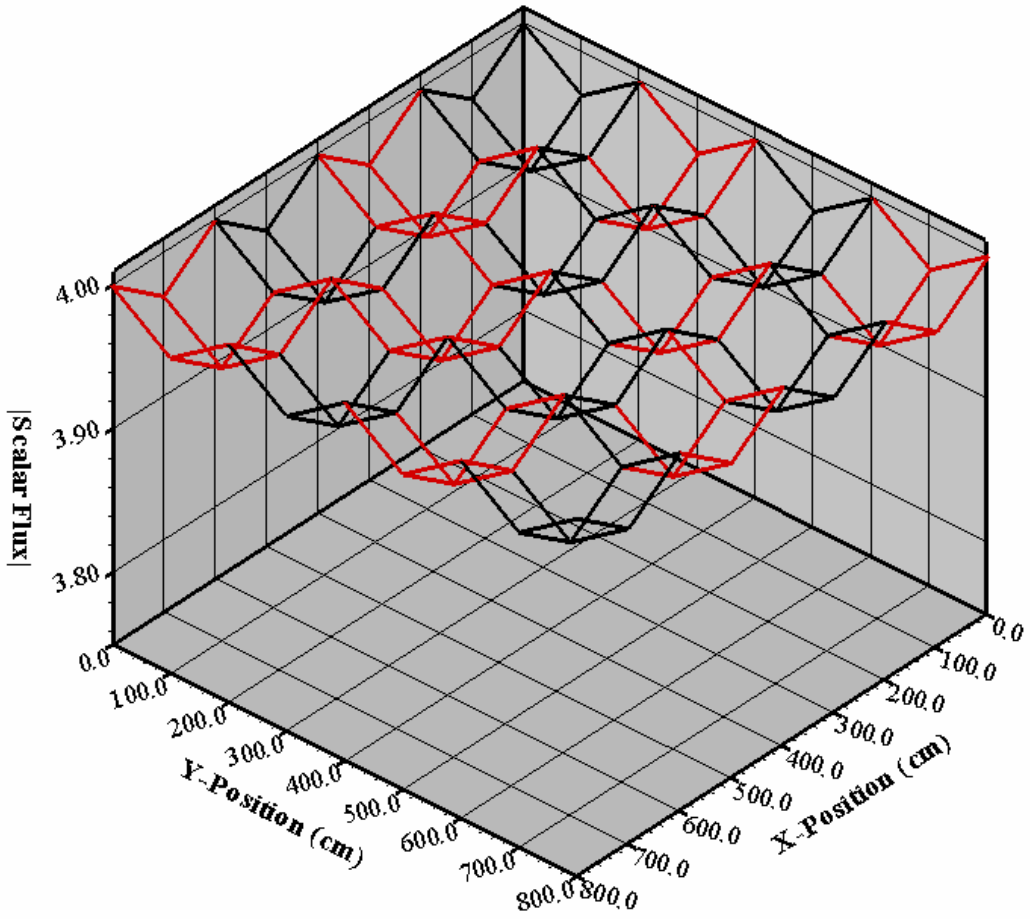

Figure 4.12 – Slowest Converging Error mode for Problem_100.0_1

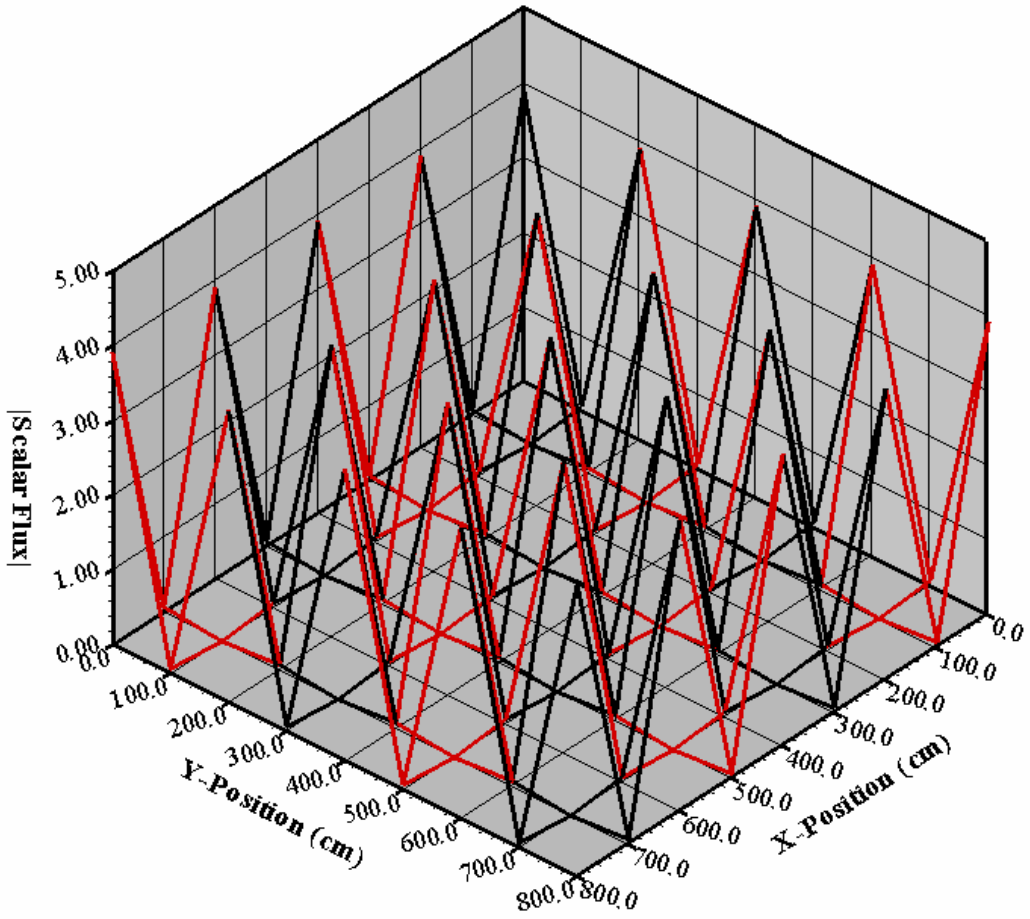

Figure 4.13 – Slowest Converging Error mode for Problem_100.0_2

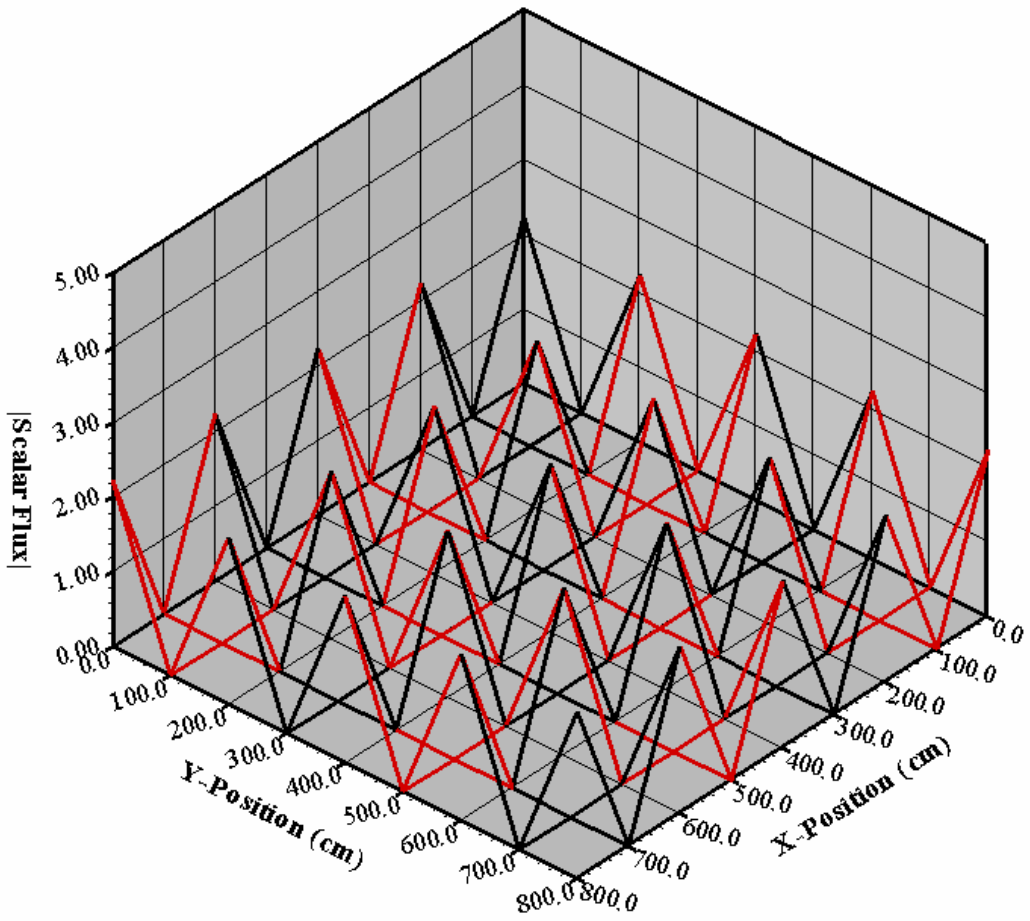

Figure 4.14 – Slowest Converging Error mode for Problem_100.0_3

Figures 4.6 – 4.14 show the spatial shape of the slowest converging error mode following simple relaxation on the fine grid. It is easily observed that as more absorption is added, the four-cell error deviates more from linearity. Interestingly, as the problems become optically thick and highly absorbing, the shape begins to deviate from the physical solution. This is not unreasonable since we are examining the eigenvectors corresponding to the largest eigenvalue and not the solution to the physical transport equations. We use these eigenvectors corresponding to the largest eigenvalues to define five (5) kink-factors. Similar to the kinked-linear interpolation of the one-dimensional

method, we define a kinked-bilinear surface using a combination of nine (9) scalar fluxes associated with the angular flux correction following relaxation:

$$\phi_1 = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^1_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^1_{(ii,jj)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^1_{(ii,jj)},mm} + \sum_{nn=1}^{N_4} w_n\psi_{K^1_{(ii,jj)},nn}, \tag{4.117}$$

$$\phi_2 = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^2_{(ii,jj+1)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^2_{(ii,jj+1)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^2_{(ii,jj+1)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^2_{(ii,jj+1)},nn}, \tag{4.118}$$

$$\phi_3 = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^3_{(ii+1,jj+1)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^3_{(ii+1,jj+1)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^3_{(ii+1,jj+1)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^3_{(ii+1,jj+1)},nn}, \tag{4.119}$$

$$\phi_4 = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^4_{(ii+1,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^4_{(ii+1,jj)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^4_{(ii+1,jj)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^4_{(ii+1,jj)},nn}, \tag{4.120}$$

$$\phi_L = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^4_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^4_{(ii,jj)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^1_{(ii+1,jj)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^1_{(ii+1,jj)},nn}, \tag{4.121}$$

$$\phi_R = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^3_{(ii,jj+1)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^3_{(ii,jj+1)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^2_{(ii+1,jj+1)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^2_{(ii+1,jj+1)},nn}, \tag{4.122}$$

$$\phi_T = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^3_{(ii+1,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^4_{(ii+1,jj+1)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^4_{(ii+1,jj+1)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^3_{(ii+1,jj)},nn}, \tag{4.123}$$

$$\phi_B = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^2_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^1_{(ii,jj+1)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^1_{(ii,jj+1)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^2_{(ii,jj)},nn}, \tag{4.124}$$

$$\phi_M = \sum_{kk=1}^{N_1} w_{kk}\psi_{K^3_{(ii,jj)},kk} + \sum_{ll=1}^{N_2} w_{ll}\psi_{K^4_{(ii,jj+1)},ll} + \sum_{mm=1}^{N_3} w_{mm}\psi_{K^1_{(ii+1,jj+1)},mm} + \sum_{nn=1}^{N_4} w_{nn}\psi_{K^2_{(ii+1,jj)},nn}. \tag{4.125}$$

Equations (4.117) - (4.125) describe nine scalar fluxes. The first four (Eqs. (4.117) - (4.120)) represent the scalar flux computed from the angular flux correction at what will become the four coarse-cell vertices. The next set of four fluxes (Eqs. (4.121) - (4.124)) represent edge scalar fluxes computed from the exiting angular fluxes at the common nodes. The final scalar flux (Eq. (4.125)) represents the scalar flux associated with the physical angular flux correction at the four-cell quad common vertex. From these nine quantities, we define five (5) kink-factors that will characterize the deviation from linearity along the edges, and the deviation from bilinearity at the common interior vertex. In the one-dimensional method, this calculation was done in mean-free-path space. This is not an option in two dimensions. As a result, we characterize the two-dimensional kink-factors in Cartesian space. To do this, we must define dimensionless parameters that represent ratios of four fine-grid-cell dimensions to one coarse-grid cell dimension:

$$\beta^x_{K_{(ii,jj)}} = \left(\frac{h^x_{K_{(ii,jj+1)}}}{h^x_{K_{(ii,jj)}} + h^x_{K_{(ii,jj+1)}}} \right), \tag{4.126}$$

$$\beta^{y}_{K_{(ii,jj)}} = \left(\frac{h^{y}_{K_{(ii+1,jj)}}}{h^{y}_{K_{(ii,jj)}} + h^{y}_{K_{(ii+1,jj)}}} \right), \tag{4.127}$$

$$\beta^{xy}_{K_{(ii,jj)}} = \left(\frac{h^{x}_{K_{(ii,jj+1)}}}{h^{x}_{K_{(ii,jj)}} + h^{x}_{K_{(ii,jj+1)}}} \right) \left(\frac{h^{y}_{K_{(ii+1,jj)}}}{h^{y}_{K_{(ii,jj)}} + h^{y}_{K_{(ii+1,jj)}}} \right), \tag{4.128}$$

$$\beta^{x}_{K_{(ii,jj+1)}} = \left(\frac{h^{x}_{K_{(ii,jj)}}}{h^{x}_{K_{(ii,jj)}} + h^{x}_{K_{(ii,jj+1)}}} \right), \tag{4.129}$$

$$\beta^{y}_{K_{(ii,jj+1)}} = \left(\frac{h^{y}_{K_{(ii+1,jj+1)}}}{h^{y}_{K_{(ii,jj+1)}} + h^{y}_{K_{(ii+1,jj+1)}}} \right), \tag{4.130}$$

$$\beta^{xy}_{K_{(ii,jj+1)}} = \left(\frac{h^{x}_{K_{(ii,jj)}}}{h^{x}_{K_{(ii,jj)}} + h^{x}_{K_{(ii,jj+1)}}} \right) \left(\frac{h^{y}_{K_{(ii+1,jj+1)}}}{h^{y}_{K_{(ii,jj+1)}} + h^{y}_{K_{(ii+1,jj+1)}}} \right), \tag{4.131}$$

$$\beta^{x}_{K_{(ii+1,jj+1)}} = \left(\frac{h^{x}_{K_{(ii+1,jj)}}}{h^{x}_{K_{(ii+1,jj+1)}} + h^{x}_{K_{(ii+1,jj)}}} \right), \tag{4.132}$$

$$\beta^{y}_{K_{(ii+1,jj+1)}} = \left(\frac{h^{y}_{K_{(ii,jj+1)}}}{h^{y}_{K_{(ii,jj+1)}} + h^{y}_{K_{(ii+1,jj+1)}}} \right), \tag{4.133}$$

$$\beta^{xy}_{K_{(ii+1,jj+1)}} = \left(\frac{h^{x}_{K_{(ii+1,jj)}}}{h^{x}_{K_{(ii+1,jj+1)}} + h^{x}_{K_{(ii+1,jj)}}} \right) \left(\frac{h^{y}_{K_{(ii,jj+1)}}}{h^{y}_{K_{(ii+1,jj+1)}} + h^{y}_{K_{(ii+1,jj+1)}}} \right), \tag{4.134}$$

$$\beta^{x}_{K_{(ii+1,jj)}} = \left(\frac{h^{x}_{K_{(ii+1,jj+1)}}}{h^{x}_{K_{(ii+1,jj+1)}} + h^{x}_{K_{(ii+1,jj)}}} \right), \tag{4.135}$$

$$\beta^{y}_{K_{(ii+1,jj)}} = \left(\frac{h^{y}_{K_{(ii,jj)}}}{h^{y}_{K_{(ii,jj)}} + h^{y}_{K_{(ii+1,jj)}}} \right), \tag{4.136}$$

$$\beta^{xy}_{K_{(ii+1,jj)}} = \left(\frac{h^x_{K_{(ii+1,jj+1)}}}{h^x_{K_{(ii+1,jj+1)}} + h^x_{K_{(ii+1,jj)}}} \right) \left(\frac{h^y_{K_{(ii,jj)}}}{h^y_{K_{(ii,jj)}} + h^y_{K_{(ii+1,jj)}}} \right). \tag{4.137}$$

It should be noted that the definitions in Eqs. (4.126) - (4.137), the beta terms can be written in terms of different cell thicknesses. Since we are using a structured rectangular grid, certain cell dimensions can be written more than one way. For example, the cell dimensions $h^x_{K_{(ii,jj)}}$ is equivalent to $h^x_{K_{(ii+1,jj)}}$. With these parameters defined, we can define the five kink-factors:

$$\alpha_L = \frac{\phi_L}{\left[\left(\beta_{K^y_{(ii,jj)}} \right) \phi_1 + \left(\beta_{K^y_{(ii+1,jj)}} \right) \phi_4 \right]}, \tag{4.138}$$

$$\alpha_R = \frac{\phi_R}{\left[\left(\beta_{K^y_{(ii,jj+1)}} \right) \phi_2 + \left(\beta_{K^y_{(ii+1,jj+1)}} \right) \phi_3 \right]}, \tag{4.139}$$

$$\alpha_T = \frac{\phi_T}{\left[\left(\beta_{K^x_{(ii+1,jj+1)}} \right) \phi_3 + \left(\beta_{K^x_{(ii+1,jj)}} \right) \phi_4 \right]}, \tag{4.140}$$

$$\alpha_B = \frac{\phi_B}{\left[\left(\beta_{K^x_{(ii,jj)}} \right) \phi_1 + \left(\beta_{K^x_{(ii,jj+1)}} \right) \phi_2 \right]}, \tag{4.141}$$

$$\alpha_M = \frac{\phi_M}{\left[\left(\beta_{K^{xy}_{(ii,jj)}} \right) \phi_1 + \left(\beta_{K^{xy}_{(ii,jj+1)}} \right) \phi_2 + \left(\beta_{K^{xy}_{(ii+1,jj+1)}} \right) \phi_3 + \left(\beta_{K^{xy}_{(ii+1,jj)}} \right) \phi_4 \right]}. \tag{4.142}$$

The five (5) kink-factors defined in Eqs. (4.138) - (4.142) define a prolongation method for mapping coarse, single cell information to four, fine cells. The assumptions that were so challenged in one-dimension are repeated here. This prolongation operator also goes into building coarse-grid operators that we think will attenuate the slowest converging error mode described in this section.

Restriction, Prolongation and Coarse-Grid Operators

The construction of restriction, prolongation and coarse-grid operators completes the formation of a multigrid iterative method. Each of these operators are constructed by requiring that they have certain desired properties. The restriction operator should preserve the zeroth and first spatial moments of the discrete equations. The prolongation operator should as accurately as possible interpolate information from coarse-grids to fine-grids. This is not necessarily the case, however, as was shown in Chapter III. Coarse-grid operators in two dimensions are defined from the restriction and prolongation operators in the same manner as the analogous operators were defined in one dimension. This construction is defined by Eq. (4.143):

$$I_{\ell}^{4\ell}\Pi^{\ell}I_{4\ell}^{\ell}=\Pi^{4\ell}. \tag{4.143}$$

The restriction operator corresponds to integrating over a four-cell quad with the finite-element weight functions. Each coarse weight function is equivalent to a cardinal basis function taking values of one (1) at its support point, varying linearly to zero along the edges of the four-cell quad, and bilinearly to zero in the interior. The initial vector is of length $16\left(N_1+N_2+N_3+N_4\right)$, and, following restriction, is of length $4\left(N_1+N_2+N_3+N_4\right)$. This leads to a restriction operator of the form in Eq. (4.144):

$$I_{\ell}^{4\ell}=\left[\begin{array}{cccc} S_{K_{(ii,jj)}} & S_{K_{(ii,jj+1)}} & S_{K_{(ii+1,jj+1)}} & S_{K_{(ii+1,jj)}} \end{array}\right], \tag{4.144}$$

where each block in Eq. (4.144) is each cell's contribution to resulting restricted vector.

The finer structure of the blocks in Eq. (4.144) is given by Eq. (4.145):

$$S_{K_{(ii,jj)}}=\left[\begin{array}{cccc} S_{K_{(ii,jj)}}^{\left(4N_1\times 4N_1\right)} & 0 & 0 & 0 \\ 0 & S_{K_{(ii,jj)}}^{\left(4N_2\times 4N_2\right)} & 0 & 0 \\ 0 & 0 & S_{K_{(ii,jj)}}^{\left(4N_3\times 4N_3\right)} & 0 \\ 0 & 0 & 0 & S_{K_{(ii,jj)}}^{\left(4N_4\times 4N_4\right)} \end{array}\right], \tag{4.145}$$

for each $K_{(ii,jj)}$ in the four-cell quad. The diagonal-block terms of the previous expression are given by Eqs. (4.146) - (4.149):

$$S_{K_{(ii,jj)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} I^{(N_i\times N_i)} & \beta_{K_{(ii,jj)}}^{x} I^{(N_i\times N_i)} & \beta_{K_{(ii,jj)}}^{xy} I^{(N_i\times N_i)} & \beta_{K_{(ii,jj)}}^{y} I^{(N_i\times N_i)} \\ 0 & \beta_{K_{(ii,jj+1)}}^{x} I^{(N_i\times N_i)} & \beta_{K_{(ii,jj+1)}}^{xy} I^{(N_i\times N_i)} & 0 \\ 0 & 0 & \beta_{K_{(ii+1,jj+1)}}^{xy} I^{(N_i\times N_i)} & 0 \\ 0 & 0 & \beta_{K_{(ii+1,jj)}}^{xy} I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj)}}^{y} I^{(N_i\times N_i)} \end{bmatrix}, \quad (4.146)$$

$$S_{K_{(ii,jj+1)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} \beta_{K_{(ii,jj)}}^{x} I^{(N_i\times N_i)} & 0 & 0 & \beta_{K_{(ii,jj)}}^{xy} I^{(N_i\times N_i)} \\ \beta_{K_{(ii,jj+1)}}^{x} I^{(N_i\times N_i)} & I^{(N_i\times N_i)} & \beta_{K_{(ii,jj+1)}}^{y} I^{(N_i\times N_i)} & \beta_{K_{(ii,jj+1)}}^{xy} I^{(N_i\times N_i)} \\ 0 & 0 & \beta_{K_{(ii+1,jj+1)}}^{y} I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj+1)}}^{xy} I^{(N_i\times N_i)} \\ 0 & 0 & \beta_{K_{(ii+1,jj)}}^{y} I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj)}}^{xy} I^{(N_i\times N_i)} \end{bmatrix}, \quad (4.147)$$

$$S_{K_{(ii+1,jj+1)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} \beta_{K_{(ii,jj)}}^{xy} I^{(N_i\times N_i)} & 0 & 0 & 0 \\ \beta_{K_{(ii,jj+1)}}^{xy} I^{(N_i\times N_i)} & \beta_{K_{(ii,jj+1)}}^{y} I^{(N_i\times N_i)} & 0 & 0 \\ \beta_{K_{(ii+1,jj+1)}}^{xy} I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj+1)}}^{y} I^{(N_i\times N_i)} & I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj+1)}}^{x} I^{(N_i\times N_i)} \\ \beta_{K_{(ii+1,jj)}}^{xy} I^{(N_i\times N_i)} & 0 & 0 & \beta_{K_{(ii+1,jj)}}^{x} I^{(N_i\times N_i)} \end{bmatrix}, \quad (4.148)$$

$$S_{K_{(ii+1,jj+1)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} \beta_{K_{(ii,jj)}}^{y} I^{(N_i\times N_i)} & \beta_{K_{(ii,jj)}}^{xy} I^{(N_i\times N_i)} & 0 & 0 \\ 0 & \beta_{K_{(ii,jj+1)}}^{xy} I^{(N_i\times N_i)} & 0 & 0 \\ 0 & \beta_{K_{(ii+1,jj+1)}}^{xy} I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj+1)}}^{x} I^{(N_i\times N_i)} & 0 \\ \beta_{K_{(ii+1,jj)}}^{y} I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj)}}^{xy} I^{(N_i\times N_i)} & \beta_{K_{(ii+1,jj)}}^{x} I^{(N_i\times N_i)} & I^{(N_i\times N_i)} \end{bmatrix}, \quad (4.149)$$

for $i=1,2,3,4$. Equations. (4.145) - (4.149) define a restriction operator that performs the operation of transferring information from four fine-grid cells to a single coarse-grid cell. Further, it does this in a manner that preserves the zeroth spatial moment (each row

of the restriction operator sums to four (4)), and the first spatial moment (each column of the restriction operator sums to one (1)).

The prolongation operator transfers information from a single coarse-grid cell to four fine-grid cells. As mentioned previously, we are interpolating information in Cartesian space and not mean-free-path space as we do in the one-dimensional case. Thus, we use the same dimensionless parameters used in the restriction operator; and we define the interpolation in terms of the five (5) kink-factors described in the previous section (Sec. IV). A prolongation operator in the two-dimensional case extends a vector of length $4\left(N_1+N_2+N_3+N_4\right)$ to length $16\left(N_1+N_2+N_3+N_4\right)$. The prolongation operator takes the form of Eq. (4.150):

$$I_{4\ell}^{\ell}=\begin{bmatrix} T_{K_{(ii,jj)}} \\ T_{K_{(ii,jj+1)}} \\ T_{K_{(ii+1,jj+1)}} \\ T_{K_{(ii+1,jj)}} \end{bmatrix}, \tag{4.150}$$

where, for each $K_{(ii,jj)}$ in the four-cell quad, the block-column terms in the previous expression are given by Eq. (4.151):

$$T_{K_{(ii,jj)}}=\begin{bmatrix} T_{K_{(ii,jj)}}^{\left(4N_1\times 4N_1\right)} & 0 & 0 & 0 \\ 0 & T_{K_{(ii,jj)}}^{\left(4N_2\times 4N_2\right)} & 0 & 0 \\ 0 & 0 & T_{K_{(ii,jj)}}^{\left(4N_3\times 4N_3\right)} & 0 \\ 0 & 0 & 0 & T_{K_{(ii,jj)}}^{\left(4N_4\times 4N_4\right)} \end{bmatrix}. \tag{4.151}$$

Each block in Eq. (4.150) represents the interpolation of the corresponding block's information from coarse, single-cell information:

$$T_{K_{(ii,jj)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha_B\beta_{K_{(ii,jj)}}^{x} & \alpha_B\beta_{K_{(ii,jj+1)}}^{x} & 0 & 0 \\ \alpha_M\beta_{K_{(ii,jj)}}^{xy} & \alpha_M\beta_{K_{(ii,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj)}}^{xy} \\ \alpha_L\beta_{K_{(ii,jj)}}^{y} & 0 & 0 & \alpha_L\beta_{K_{(ii+1,jj)}}^{y} \end{bmatrix}, \tag{4.152}$$

$$T_{K_{(ii,jj+1)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} \alpha_B\beta_{K_{(ii,jj)}}^{x} & \alpha_B\beta_{K_{(ii,jj+1)}}^{x} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \alpha_R\beta_{K_{(ii,jj+1)}}^{y} & \alpha_R\beta_{K_{(ii+1,jj+1)}}^{y} & 0 \\ \alpha_M\beta_{K_{(ii,jj)}}^{xy} & \alpha_M\beta_{K_{(ii,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj)}}^{xy} \end{bmatrix}, \tag{4.153}$$

$$T_{K_{(ii+1,jj+1)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} \alpha_M\beta_{K_{(ii,jj)}}^{xy} & \alpha_M\beta_{K_{(ii,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj)}}^{xy} \\ 0 & \alpha_R\beta_{K_{(ii,jj+1)}}^{y} & \alpha_R\beta_{K_{(ii+1,jj+1)}}^{y} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \alpha_T\beta_{K_{(ii+1,jj+1)}}^{x} & \alpha_T\beta_{K_{(ii+1,jj)}}^{x} \end{bmatrix}, \tag{4.154}$$

$$T_{K_{(ii+1,jj)}}^{(4N_i\times 4N_i)}=\begin{bmatrix} \alpha_L\beta_{K_{(ii,jj)}}^{y} & 0 & 0 & \alpha_L\beta_{K_{(ii+1,jj)}}^{y} \\ \alpha_M\beta_{K_{(ii,jj)}}^{xy} & \alpha_M\beta_{K_{(ii,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj+1)}}^{xy} & \alpha_M\beta_{K_{(ii+1,jj)}}^{xy} \\ 0 & 0 & \alpha_T\beta_{K_{(ii+1,jj+1)}}^{x} & \alpha_T\beta_{K_{(ii+1,jj)}}^{x} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.155}$$

for $i=1,2,3,4$. It must be noted that each entry in Eqs. (4.152) - (4.155) is scaled by the identity matrix of dimension $\left(N_i\times N_i\right)$. This has been left out of the notation for brevity. Eqs. (4.150) - (4.155) describe a kinked-bilinear prolongation operator. As in the one-dimensional case, there are tacit assumptions that will compromise the accuracy of the prolongation – continuity, and isotropy. We know that continuity is easily violated, and isotropy will most certainly be compromised. Recall the one-dimensional, heterogeneous cases where angular shapes can be quite different. Numerical tests of these assumptions are presented in the next chapter (Chapter V).

While Eq. (4.143) tells us how to build coarse-grid operators, there is an additional concern with this operation for the two-dimensional case. The derivation and implementation of the one-dimensional method was performed without consideration for data storage space. In the two-dimensional case, this could easily become a limiting factor if care is not take to store only what is necessary without sacrificing generality. Our implementation of the BLDFEM equations allows every cell to have unique properties. These properties, in turn, define coefficients that scale an identity matrix, a diagonal array of directional cosines, or an array of quadrature weights. With this in mind, we derive coarse-grid operators that are designed to attenuate the dominate error mode on a coarser grid and require a minimum of storage space without sacrificing generality.

In block-matrix form, there are five (5) operators, per cell, that define the complete S_N transport equation that describes a given problem: $\Pi^{\ell}_{K_{(ii,jj)}}$, $\Theta^{\ell}_{K_{(ii,jj)}}$, $\Gamma^{\ell}_{K_{(ii,jj)}}$, $A^{\ell}_{K_{(ii,jj)}}$ and $B^{\ell}_{K_{(ii,jj)}}$. Each of these five (5) operators is also comprised of finer block structures (see Sec. II). The finer block structure further reveals a point critical to data-storage issues. It is easy to see that each of the boundary operators on the fine grid, while dimensionally large, is described by a total of four (4) numbers each. At this point we have no reason to anticipate that this will also be true on coarser grid levels; however, with the one-dimensional experience in mind, there is reason to believe that our assumptions in the prolongation operator will have a major part to play in the result. Regardless, on the fine grid we can see that a cell's boundary can be completely described by sixteen (16) numbers regardless of quadrature order. A similar, yet slightly more involved, observation can be made about the single-cell transport operator. Equations (4.37) - (4.42) show that each streaming operator can be described by, at most, sixteen (16) numbers each. Thus, all streaming information can be described by sixty-four (64) numbers per cell – sixteen per angular half-space. In a similar manner, we observe that the collision-rate-density and scattering-rate-density operators are also described by at most sixteen (16) coefficients. Thus, all collision-rate-density and

scattering-rate-density information can be described by thirty-two (32) numbers. Our implementation also stores cell dimensions with this information resulting in a storage demand of one hundred and fourteen (114) double-precision-real (i.e., eight (8) bytes) values per cell. The critical point is to recognize that this storage requirement does not scale with quadrature order. As a result, we can scale the quadrature order for a given problem and, necessarily, the storage requirement for its solution vector; however, we will have to store no further information to completely describe the transport operators. The next thing to show is that this storage requirement is true for every cell at every grid level.

We begin by coarsening the block-matrix four-cell operator:

$$\Pi^{4\ell}_{K_{(ii,jj)}} = I^{4\ell}_{\ell} \begin{bmatrix} \Pi^{\ell}_{K_{(ii,jj)}} & -\Theta^{\ell}_{K_{(ii,jj)}} & 0 & -\mathrm{A}^{\ell}_{K_{(ii,jj)}} \\ -\Gamma^{\ell}_{K_{(ii,jj+1)}} & \Pi^{\ell}_{K_{(ii,jj+1)}} & -\mathrm{A}^{\ell}_{K_{(ii,jj+1)}} & 0 \\ 0 & -\mathrm{B}^{\ell}_{K_{(ii+1,jj+1)}} & \Pi^{\ell}_{K_{(ii+1,jj+1)}} & -\Gamma^{\ell}_{K_{(ii+1,jj+1)}} \\ -\mathrm{B}^{\ell}_{K_{(ii,jj+1)}} & 0 & -\Theta^{\ell}_{K_{(ii+1,jj)}} & \Pi^{\ell}_{K_{(ii+1,jj)}} \end{bmatrix} I^{\ell}_{4\ell} \,. \tag{4.156}$$

After a bit of stimulating algebra, the operation defined in Eq. (4.156) results in Eq. (4.157) – the coarse-grid operator that has been constructed from all of the information on the fine grid associated with the four-cell quad:

$$\Pi^{4\ell}_{K_{(ii,jj)}} = \left\{ \tilde{\Pi}^{4\ell}_{K_{(ii,jj)}} - \tilde{\Gamma}^{4\ell}_{K_{(ii,jj)}} - \tilde{\mathrm{A}}^{\ell}_{K_{(ii,jj)}} - \tilde{\mathrm{B}}^{\ell}_{K_{(ii,jj)}} - \tilde{\Theta}^{4\ell}_{K_{(ii,jj)}} \right\}, \tag{4.157}$$

where:

$$\tilde{\Pi}^{4\ell}_{K_{(ii,jj)}} = \sum_{k}^{\{(ii,jj),(ii,jj+1),(ii+1,jj+1),(ii+1,jj)\}} \left(S\right)_{K_k} \Pi^{\ell}_{K_k} \left(T\right)_{K_k} , \tag{4.158}$$

$$\tilde{\Gamma}^{4\ell}_{K_{(ii,jj)}} = \left[\left(\underline{S}\right)_{K_{(ii,jj+1)}} \underline{\Gamma}^{\ell}_{K_{(ii,jj+1)}} \left(\underline{T}\right)_{K_{(ii,jj)}} + \left(\underline{S}\right)_{K_{(ii+1,jj+1)}} \underline{\Gamma}^{\ell}_{K_{(ii+1,jj+1)}} \left(\underline{T}\right)_{K_{(ii+1,jj)}} \right], \tag{4.159}$$

$$\tilde{\mathrm{A}}^{4\ell}_{K_{(ii,jj)}} = \left[\left(S\right)_{K_{(ii,jj+1)}} \mathrm{A}^{\ell}_{K_{(ii,jj+1)}} \left(T\right)_{K_{(ii+1,jj+1)}} + \left(S\right)_{K_{(ii,jj)}} \mathrm{A}^{\ell}_{K_{(ii,jj)}} \left(T\right)_{K_{(ii+1,jj)}} \right], \tag{4.160}$$

$$\tilde{\mathrm{B}}^{4\ell}_{K_{(ii,jj)}} = \left[\left(S\right)_{K_{(ii+1,jj+1)}} \mathrm{B}^{\ell}_{K_{(ii+1,jj+1)}} \left(T\right)_{K_{(ii,jj+1)}} + \left(S\right)_{K_{(ii+1,jj)}} \mathrm{B}^{\ell}_{K_{(ii+1,jj)}} \left(T\right)_{K_{(ii,jj)}} \right], \tag{4.161}$$

$$\tilde{\Theta}^{4\ell}_{K_{(ii,jj)}} = \left[(S)_{K_{(ii+1,jj)}} \Theta^{\ell}_{K_{(ii+1,jj)}} (T)_{K_{(ii+1,jj+1)}} + (S)_{K_{(ii,jj)}} \Theta^{\ell}_{K_{(ii,jj)}} (T)_{K_{(ii,jj+1)}} \right]. \qquad (4.162)$$

As in the one-dimensional case, each of the original four cell's streaming-rate-density, collision-rate-density and scattering-rate-density contributes to the analogous information on the next grid. The fine-grid boundary-communication operators become part of the streaming-rate-density operators on the coarse grid, losing their original physical interpretation. After a bit of thought, we can now see that the storage requirements of the fine-grid cells are the same for the coarse-grid cells. This can be observed from Eqs. (4.158) - (4.162) by noting that each operator is multiplied on the left by a restriction operator which has no angularly dependent information, and on the right by a prolongation operator that has kink-factors that are not allowed to vary with angle. Consequently, every cell can have all transport properties defined by one-hundred and fourteen (114) numbers at each grid level. This is a significant savings in machine memory.

The remaining operators to define on coarser grids are the boundary communication operators. Some of these represent a shift in the operator's physical interpretation. On the fine grid, these operators represented information communication across the boundaries of four-cell quads. On the next coarser grid, they represent intra-quad communication. We paid particular attention to the analogous calculation in one dimension to be absolutely certain that operator structure did not change from grid level to grid level. If this had been the case, it would have a great impact on how the method is implemented. With most complicated boundary communication operators in two-dimensions, coupled with the storage concerns already presented, we take even greater care to assure that the same structure (and, therefore, function) and storage requirement for these operators on the fine grid is the same for all coarser grids. Following a significant amount of work, we determine that these requirements are indeed met. This development is further complicated by the fact that each boundary communication term appears twice in the four-cell operator. It is not surprising that great care must be taken with how these operators are built. If this method is applied to heterogeneous or non-uniform grid problems, these operators can look very different from cell to cell.

Fortunately, a pattern emerges that allows us to give one example of each operator. The coarse boundary communication that communicated four-cell quads to other four-cell quads on one grid, but perform the function of inter-quad communication on the next coarser grid are given by Eqs. (4.163) - (4.166):

$$\Gamma^{4\ell}_{K_{(ii,jj+1)}} = (S)_{K_{(ii,jj+2)}} \Gamma^{\ell}_{K_{(ii,jj+2)}} (T)_{K_{(ii,jj+1)}} + (S)_{K_{(ii+1,jj+2)}} \Gamma^{\ell}_{K_{(ii+1,jj+2)}} (T)_{K_{(ii+1,jj+1)}}, \quad (4.163)$$

$$\Theta^{4\ell}_{K_{(ii,jj)}} = (S)_{K_{(ii,jj+1)}} \Theta^{\ell}_{K_{(ii,jj+1)}} (T)_{K_{(ii,jj+2)}} + (S)_{K_{(ii+1,jj+1)}} \Theta^{\ell}_{K_{(ii+1,jj+1)}} (T)_{K_{(ii+1,jj+2)}}, \quad (4.164)$$

$$\mathrm{A}^{4\ell}_{K_{(ii,jj+1)}} = (S)_{K_{(ii+1,jj+3)}} \mathrm{A}^{\ell}_{K_{(ii+1,jj+3)}} (T)_{K_{(ii+2,jj+3)}} + (S)_{K_{(ii+1,jj+2)}} \mathrm{A}^{\ell}_{K_{(ii+1,jj+2)}} (T)_{K_{(ii+2,jj+2)}}, \quad (4.165)$$

$$\mathrm{B}^{4\ell}_{K_{(ii+1,jj)}} = (S)_{K_{(ii+2,jj)}} \mathrm{B}^{\ell}_{K_{(ii+2,jj)}} (T)_{K_{(ii+1,jj)}} + (S)_{K_{(ii+2,jj+1)}} \mathrm{B}^{\ell}_{K_{(ii+2,jj+1)}} (T)_{K_{(ii+1,jj+1)}}. \quad (4.166)$$

The remaining boundary communication terms represent those boundary communication terms that communicated four-cell quad boundary information and on the next coarser grid perform the same function. This is distinctly different from the previous set of operators presented. This type of comparison again reassures us that our storage minimization efforts will apply to all operators on all grids levels. Again, a pattern emerges and we can give one example of each operator. This final set of coarse-grid operators are given by Eqs. (4.167) - (4.170):

$$\Gamma^{4\ell}_{K_{(ii+1,jj+2)}} = (S)_{K_{(ii+2,jj+4)}} \Gamma^{\ell}_{K_{(ii+2,jj+4)}} (T)_{K_{(ii+2,jj+3)}} + (S)_{K_{(ii+3,jj+4)}} \Gamma^{\ell}_{K_{(ii+3,jj+4)}} (T)_{K_{(ii+3,jj+3)}}, \quad (4.167)$$

$$\Theta^{4\ell}_{K_{(ii,jj+1)}} = (S)_{K_{(ii,jj+3)}} \Theta^{\ell}_{K_{(ii,jj+3)}} (T)_{K_{(ii,jj+4)}} + (S)_{K_{(ii+1,jj+3)}} \Theta^{\ell}_{K_{(ii+1,jj+3)}} (T)_{K_{(ii+1,jj+4)}}, \quad (4.168)$$

$$\mathrm{A}^{4\ell}_{K_{(ii+1,jj+1)}} = (S)_{K_{(ii+3,jj+3)}} \mathrm{A}^{\ell}_{K_{(ii+3,jj+3)}} (T)_{K_{(ii+4,jj+3)}} + (S)_{K_{(ii+3,jj+2)}} \mathrm{A}^{\ell}_{K_{(ii+3,jj+2)}} (T)_{K_{(ii+4,jj+2)}}, \quad (4.169)$$

$$\mathrm{B}^{4\ell}_{K_{(ii+2,jj+2)}} = (S)_{K_{(ii+4,jj+4)}} \mathrm{B}^{\ell}_{K_{(ii+4,jj+4)}} (T)_{K_{(ii+3,jj+4)}} + (S)_{K_{(ii+4,jj+5)}} \mathrm{B}^{\ell}_{K_{(ii+4,jj+5)}} (T)_{K_{(ii+3,jj+5)}}. \quad (4.170)$$

Equations (4.163) - (4.170), while differing in the individual operators involved, require the same type of operations. The two-dimensional problem clearly shows how each cell on one grid contributes to the coarse grid problem. Further, these operations are computationally trivial. At no time are two matrices larger than 4×4 multiplied – this is also true for all other coarse-operator types. At no time are more than four numbers

needed to describe boundary communication operators regardless of their specific function, grid level or the quadrature order. With these operators defined, we have completely described how to build the two-dimensional coarse problem without loss of generality and keeping machine storage to a minimum.

Summary of Chapter IV

In this chapter, we presented a family of BLDFEMs to discretize the spatial domain. They exhibit the same type of desirable behavior that the one-dimensional LDFEMs exhibit and which led to the excellent convergence behavior observed by Manteuffel and verified by our work for a specific problem class (Chapter III) [35]. We have shown that a complete BLDFEM, S_N transport problem can be written in block matrix form – more complicated, but in the same spirit as Manteuffel's original idea. This system of equations can be decomposed into the form required by Sherman-Morrison for efficient inversion. While these are important components that affect numerical implementation, they are simply procedural. The critical point in the development of this method, as it was in one dimension, is the ability to characterize error following relaxation and using that characterization to approximate fine grid information from coarse gird information. This marks a significant deviation from the one-dimensional methodology. The type of algebraic analysis that provided Manteuffel the motivation for pursuing his original work is not tractable in two dimensions. In place of this analysis, we have shown that our Fourier analysis can be used to characterize the slowest converging error mode. This characterization is consolidated into the computation of five kink-factors that then define a prolongation operator. A restriction operator was presented that ensures we capture the zeroth and first spatial moments of the transport equation – this is precisely the same features that motivated the analogous development in one dimension. Using the restriction and prolongation operators, coarse-grid operators are constructed in the same manner as in one-dimension. Again, this is a procedural step; however, we have shown that the storage minimization effort that holds

for the fine grid also holds for all coarser grids. The elements in this chapter lead to the implementation of our new, two-dimensional multigrid solver for S_N transport problems.

In the next chapter, Chapter V, we present numerical results of this implementation. We proceed with the identical motivation that we have in Chapter III – implement, test and analyze the two-dimensional multigrid method. The implementation consists of a general S_N solver, S_NAPPER_2D, and the two-dimensional extension of the Fourier analysis of a three-level multigrid iteration matrix on an infinite periodic problem.

CHAPTER V

TWO-DIMENSIONAL NUMERICAL RESULTS

Introduction to Chapter V

Chapter IV presents the theoretical foundation and the practical details of implementing our two-dimensional spatial multigrid iterative method. In this chapter, we present a discussion of those aspects of our Fourier analysis that are unique to the two-dimensional problem, results of a two-dimensional test-suite, and an evaluation of this method as a candidate for a Krylov-solver preconditioner.

64	63	60	59	48	47	44	43
61	62	57	58	45	46	41	42
52	51	56	55	36	35	40	39
49	50	53	54	33	34	37	38
16	15	12	11	32	31	28	27
13	14	9	10	29	30	25	26
4	3	8	7	20	19	24	23
1	2	5	6	17	18	21	22

Figure 5.1 – Cell-Operator Location in the Global Block-Diagonal Structure

The choice of the four-cell relaxation step detailed in Chapter IV has an added, but not required, benefit. For the eight-by-eight problem detailed in Figure 5.1, the resulting iteration matrix, in operator form, has the same structure as in the one-dimensional case:

$$(\omega)\underline{a}^{\ell} = [A_1 - A_2 - A_3]\underline{a}^{\ell}, \tag{5.1}$$

where

$$A_1 = \left[\left(L^{\ell}\right)^{-1}\tilde{R}^{\ell}\right]^{(\nu_1+\nu_2)}, \tag{5.2}$$

$$\begin{aligned} A_2 = &\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_2} I_{4\ell}^{\ell}\left\{\left\{\sum_{l=0}^{(\nu_1+\nu_2)}\left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{l}\right\} - \left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{(\nu_1+\nu_2)}\right\} \\ &\left(L^{4\ell}\right)^{-1} I_{\ell}^{4\ell}\left(L^{\ell} - \tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_1}, \end{aligned} \tag{5.3}$$

and, if the coarsest grid is solved exactly:

$$\begin{aligned} A_3 = &\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_2} I_{4\ell}^{\ell}\left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{\nu_2} I_{16\ell}^{4\ell}\left(L^{16\ell} - \tilde{R}^{16\ell}\right)^{-1} I_{4\ell}^{16\ell} \\ &\left\{\left(L^{4\ell}\right)\left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{\nu_1}\left(L^{4\ell}\right)^{-1}\right\} I_{\ell}^{4\ell}\left(L^{\ell} - \tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_1}, \end{aligned} \tag{5.4}$$

or, if further relaxations are performed at the coarsest grid-level:

$$\begin{aligned} A_3 = &\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_2} I_{4\ell}^{\ell}\left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{\nu_2} I_{16\ell}^{4\ell} \\ &\left\{\left\{\sum_{l=0}^{(\nu_1+\nu_2)}\left[\left(L^{16\ell}\right)^{-1}\left(\tilde{R}^{16\ell}\right)\right]^{l}\right\} - \left[\left(L^{16\ell}\right)^{-1}\left(\tilde{R}^{16\ell}\right)\right]^{(\nu_1+\nu_2)}\right\}\left(L^{16\ell}\right)^{-1} I_{4\ell}^{16\ell} \\ &\left\{\left(L^{4\ell}\right)\left[\left(L^{4\ell}\right)^{-1}\left(\tilde{R}^{4\ell}\right)\right]^{\nu_1}\left(L^{4\ell}\right)^{-1}\right\} I_{\ell}^{4\ell}\left(L^{\ell} - \tilde{R}^{\ell}\right)\left[\left(L^{\ell}\right)^{-1}\left(\tilde{R}^{\ell}\right)\right]^{\nu_1}. \end{aligned} \tag{5.5}$$

The details of constructing the individual operators in Eqs. (5.1) - (5.5) are presented in the following section.

Fourier Analysis of a Two-Dimensional Multigrid Iterative Method

Constructing the global iteration matrix for the one-dimensional case is simple. Sequentially numbering the eight cells from left-right leads, automatically, to a block-diagonal structure of the global iteration matrix. The two-dimensional problem does not provide an analogous simple indexing that keeps the main blocks on the diagonal. A key lesson from the one-dimensional multigrid structure, and Fourier analysis, is that given a fine-grid operator in block-diagonal form, all coarse-grid operators will also be in block-diagonal form. In Chapter IV there was no need to consider the global structure of the iteration matrix, because it is not necessary; however, we must confront this issue for the two-dimensional Fourier analysis. Figure 5.1 shows a cyclic, counter-clockwise cell ordering used to assure that the block-diagonal structure on the fine grid is preserved on all coarser grids. The global iteration matrix is far too large to be presented in its entirety; however, using sufficiently nested notation, the main points are presentable.

To mimic an infinite medium, we impose periodic boundary conditions on the 64-cell problem in Figure 5.1. The horizontal boundary conditions are given by Eqs. (5.6) and (5.7):

$$\underline{\Psi}^{\ell}_{K_{(ii,1)}} = e^{-i\lambda_x h_x}\Gamma^{\ell}_{K_{(ii,1)}}\underline{\Psi}^{\ell}_{K_{(ii,8)}}, \tag{5.6}$$

$$\underline{\Psi}^{\ell}_{K_{(ii,8)}} = e^{i\lambda_x h_x}\Theta^{\ell}_{K_{(ii,1)}}\underline{\Psi}^{\ell}_{K_{(ii,1)}}, \tag{5.7}$$

where h_x represents the horizontal width of the entire problem, and $ii \in [1,8]$. The vertical boundary conditions are given by (5.8) and (5.9):

$$\underline{\Psi}^{\ell}_{K_{(1,jj)}} = e^{-i\lambda_y h_y}\mathrm{A}^{\ell}_{K_{(1,jj)}}\underline{\Psi}^{\ell}_{K_{(8,jj)}}, \tag{5.8}$$

$$\underline{\Psi}^{\ell}_{K_{(8,jj)}} = e^{i\lambda_y h_y}\mathrm{B}^{\ell}_{K_{(8,jj)}}\underline{\Psi}^{\ell}_{K_{(1,jj)}}, \tag{5.9}$$

where h_y represents the vertical width of the entire problem, and $jj \in [1,8]$. Since the width of the problem is represented in the exponents, the Fourier wave-number pairs, (λ_x, λ_y), must have units of inverse-length:

$$0 \le \lambda_x < \frac{2\pi}{h_x}, \tag{5.10}$$

$$0 \le \lambda_y < \frac{2\pi}{h_y}. \tag{5.11}$$

Note that in Eqs. (5.10) and (5.11), only one endpoint of the wave-number range is included. This is an important point related to a major implementation difference between the one-dimensional and two-dimensional Fourier analysis codes.

The one-dimensional global iteration matrix has dimension (32×32), but it has only eight (8) non-zero eigenvalues. Thus, if one-thousand (1,000) wave numbers are evaluated, there are eight thousand (8,000) distinct eigenvalues that will determine the suitability of this multigrid iterative method to precondition a Krylov solver. The two-dimensional iteration matrix is (1024×1024) and has four-hundred forty-eight (448) non-zero eigenvalues. If we attempt to evaluate (1000×1000) wave number pairs, we will be computing 4.48×10^8 eigenvalues and corresponding eigenvectors. This is a massive computation. To reduce the wall-clock time required to evaluate the two-dimensional global iteration matrix, we have implemented the two-dimensional Fourier analysis code using a parallel computing framework.

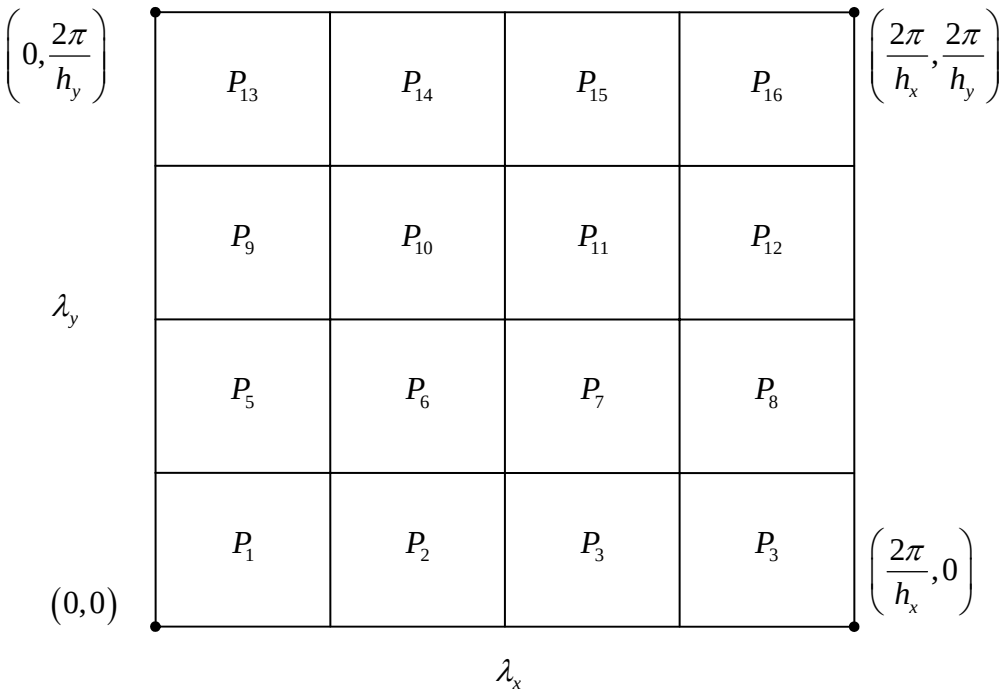

Figure 5.2 – Example of a 16-Processor Fourier-Domain Decomposition

Figure 5.2 shows how the domain of Fourier wave-number pairs are decomposed and assigned to different processors. Figure 5.3 shows how repetitive calculations are avoided by not allowing processor boundaries to share wave-number pairs. Since each Fourier error mode is an independent calculation, this parallelization is particularly simple. There is no processor-to-processor communication; thus, good processor scaling can be expected. This parallel analysis code has been implemented on the Compaq QSC massively-parallel machine at Los Alamos National Laboratory.

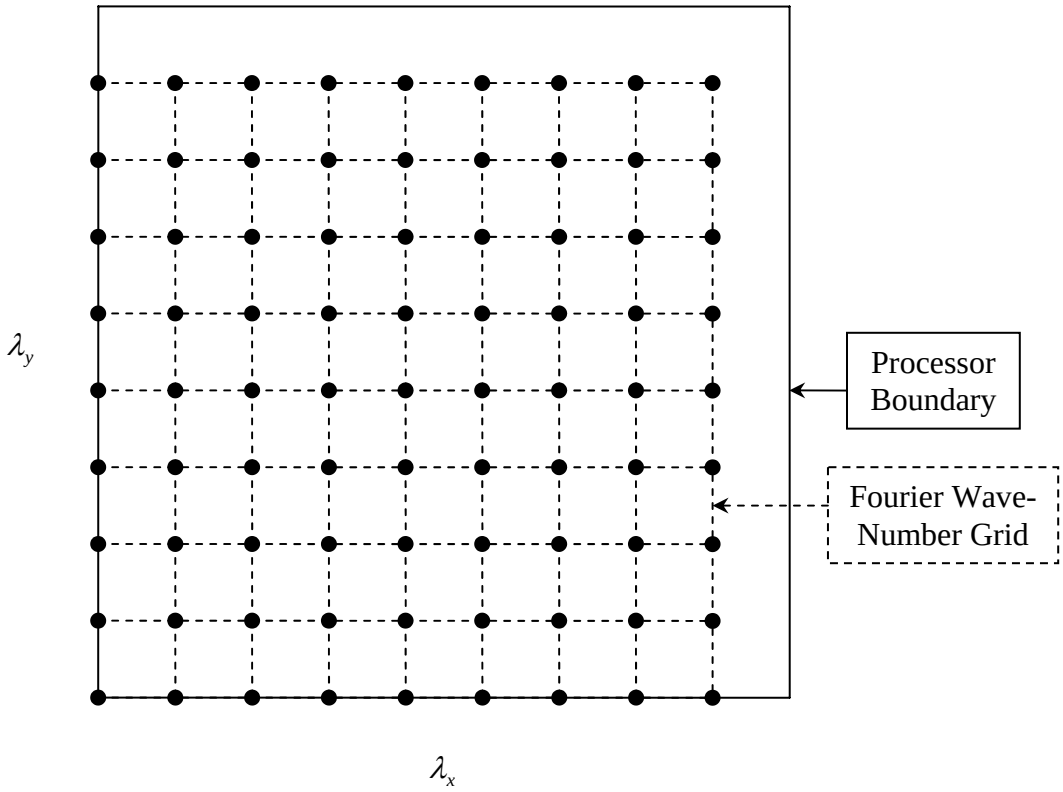

Figure 5.3 – Fourier Wave-Number Grid on a Single Processor

A detailed presentation of the complete two-dimensional Fourier iteration matrix is unnecessary since such a presentation would be repetitive; however, we can write the iteration matrix in block-quad-matrix form. Figure 5.4 shows four-quad operators along the block-diagonal, quad-to-quad boundary communication terms and the Fourier boundary communication terms. Each of these operators, Eqs. (5.12) - (5.16), contains cellwise information:

$$\begin{bmatrix}
\underline{\Pi}_1^\ell & -\underline{\Theta}_1^\ell & 0 & -\underline{A}_1^\ell & 0 & -\underline{\Gamma}_1^\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\underline{B}_1^\ell \\
-\underline{\Gamma}_2^\ell & \underline{\Pi}_2^\ell & -\underline{A}_2^\ell & 0 & -\underline{\Theta}_2^\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\underline{B}_2^\ell & 0 \\
0 & -\underline{B}_3^\ell & \underline{\Pi}_3^\ell & -\underline{\Gamma}_3^\ell & 0 & 0 & 0 & -\underline{\Theta}_3^\ell & 0 & 0 & 0 & 0 & 0 & -\underline{A}_3^\ell & 0 & 0 \\
-\underline{B}_4^\ell & 0 & -\underline{\Theta}_4^\ell & \underline{\Pi}_4^\ell & 0 & 0 & -\underline{\Gamma}_4^\ell & 0 & 0 & 0 & 0 & 0 & -\underline{A}_4^\ell & 0 & 0 & 0 \\
0 & -\underline{\Gamma}_5^\ell & 0 & 0 & \underline{\Pi}_5^\ell & -\underline{\Theta}_5^\ell & 0 & -\underline{A}_5^\ell & 0 & 0 & 0 & -\underline{B}_5^\ell & 0 & 0 & 0 & 0 \\
-\underline{\Theta}_6^\ell & 0 & 0 & 0 & -\underline{\Gamma}_6^\ell & \underline{\Pi}_6^\ell & -\underline{A}_6^\ell & 0 & 0 & 0 & -\underline{B}_6^\ell & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\underline{\Theta}_7^\ell & 0 & -\underline{B}_7^\ell & \underline{\Pi}_7^\ell & -\underline{\Gamma}_7^\ell & 0 & -\underline{A}_7^\ell & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -\underline{\Gamma}_8^\ell & 0 & -\underline{B}_8^\ell & 0 & -\underline{\Theta}_8^\ell & \underline{\Pi}_8^\ell & -\underline{A}_8^\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\underline{B}_9^\ell & \underline{\Pi}_9^\ell & -\underline{\Theta}_9^\ell & 0 & -\underline{A}_9^\ell & 0 & -\underline{\Gamma}_9^\ell & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\underline{B}_{10}^\ell & 0 & -\underline{\Gamma}_{10}^\ell & \underline{\Pi}_{10}^\ell & -\underline{A}_{10}^\ell & 0 & -\underline{\Theta}_{10}^\ell & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -\underline{A}_{11}^\ell & 0 & 0 & 0 & -\underline{B}_{11}^\ell & \underline{\Pi}_{11}^\ell & -\underline{\Gamma}_{11}^\ell & 0 & 0 & 0 & -\underline{\Theta}_{11}^\ell \\
0 & 0 & 0 & 0 & -\underline{A}_{12}^\ell & 0 & 0 & 0 & -\underline{B}_{12}^\ell & 0 & -\underline{\Theta}_{12}^\ell & \underline{\Pi}_{12}^\ell & 0 & 0 & -\underline{\Gamma}_{12}^\ell & 0 \\
0 & 0 & 0 & -\underline{B}_{13}^\ell & 0 & 0 & 0 & 0 & 0 & -\underline{\Gamma}_{13}^\ell & 0 & 0 & \underline{\Pi}_{13}^\ell & -\underline{\Theta}_{13}^\ell & 0 & -\underline{A}_{13}^\ell \\
0 & 0 & -\underline{B}_{14}^\ell & 0 & 0 & 0 & 0 & 0 & -\underline{\Theta}_{14}^\ell & 0 & 0 & 0 & -\underline{\Gamma}_{14}^\ell & \underline{\Pi}_{14}^\ell & -\underline{A}_{14}^\ell & 0 \\
0 & -\underline{A}_{15}^\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\underline{\Theta}_{15}^\ell & 0 & -\underline{B}_{15}^\ell & \underline{\Pi}_{15}^\ell & -\underline{\Gamma}_{15}^\ell \\
-\underline{A}_{16}^\ell & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\underline{\Gamma}_{16}^\ell & 0 & -\underline{B}_{16}^\ell & 0 & -\underline{\Theta}_{16}^\ell & \underline{\Pi}_{16}^\ell
\end{bmatrix}$$

Figure 5.4- Visualization of the Global Fourier Iteration Matrix

$$\underline{\Pi}_1^\ell = \begin{bmatrix}
\Pi_{K_{(1,1)}}^\ell & -\Theta_{K_{(1,1)}}^\ell & 0 & -\mathrm{A}_{K_{(1,1)}}^\ell \\
-\Gamma_{K_{(1,2)}}^\ell & \Pi_{K_{(1,2)}}^\ell & -\mathrm{A}_{K_{(1,2)}}^\ell & 0 \\
0 & -\mathrm{B}_{K_{(2,2)}}^\ell & \Pi_{K_{(2,2)}}^\ell & -\Gamma_{K_{(2,2)}}^\ell \\
-\mathrm{B}_{K_{(2,1)}}^\ell & 0 & -\Theta_{K_{(2,1)}}^\ell & \Pi_{K_{(2,1)}}^\ell
\end{bmatrix}, \tag{5.12}$$

$$\underline{\Theta}_1^\ell = \begin{bmatrix}
0 & 0 & 0 & 0 \\
\Theta_{K_{(1,2)}}^\ell & 0 & 0 & 0 \\
0 & 0 & 0 & \Theta_{K_{(2,2)}}^\ell \\
0 & 0 & 0 & 0
\end{bmatrix}, \tag{5.13}$$

$$\underline{\mathrm{A}}_1^{\ell} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \mathrm{A}_{K_{(2,2)}}^{\ell} & 0 & 0 \\ \mathrm{A}_{K_{(2,1)}}^{\ell} & 0 & 0 & 0 \end{bmatrix}, \tag{5.14}$$

$$\underline{\Gamma}_1^{\ell} = \begin{bmatrix} 0 & \Gamma_{K_{(1,1)}}^{\ell} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \Gamma_{K_{(2,1)}}^{\ell} & 0 \end{bmatrix}, \tag{5.15}$$

$$\underline{\mathrm{B}}_1^{\ell} = \begin{bmatrix} 0 & 0 & 0 & \mathrm{B}_{K_{(1,1)}}^{\ell} \\ 0 & 0 & \mathrm{B}_{K_{(1,2)}}^{\ell} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{5.16}$$

Each of the block-operators in Eqs. (5.12) - (5.16) is defined in Chapter IV, but all operators in the two-dimensional (and one-dimensional) Fourier analysis are restricted to the S_2 form. Because of the numbering scheme detailed in Figure 5.1, coarse-grid operators will be of the same form as in Eqs. (5.12) - (5.16).

The described Fourier analysis is used to buttress the numerical results obtained from the two-dimensional, S_N, multigrid method described in Chapter IV and implemented in S_NAPPER_2D. We use the one-dimensional template of four problem-classes as the test suite. For each problem, we present Fourier S_2 results and S_NAPPER_2D results for S_2, S_4, S_8, and S_{16} quadratures. We present results for both LBLD $\left(\theta_x = \theta_y = 1.0\right)$ and standard BLD $\left(\theta_x = \theta_y = 3.0\right)$. The Fourier results represent an infinite medium, and the S_NAPPER_2D results are run with an 8×8, 64-cell repeated pattern resulting in a total problem size of 128×128 cells.

Two-Dimensional Numerical Results – Homogeneous Material and Uniform Grid Problems

For this class of problems, we construct a test suite similar to the one-dimensional test-suite. The problems represent a range of cell-thickness, scattering ratios and quadrature orders as well as discretization coefficients. We examine a range of cell thicknesses spanning eight (8) orders of magnitude:

$$1.0 \le 10^{-4} \le \sigma_t h_x = \sigma_t h_y \le 1.0 \le 10^4 , \tag{5.17}$$

in increments of an order of magnitude. Scattering ratios vary from purely absorbing to purely scattering:

$$0.0 \le c \le 1.0 , \tag{5.18}$$

in the same manner as the one-dimensional test suite (Chapter III). Our first set of results will use a unit, isotropic boundary condition for the computation of the five (5) kink-factors while executing a $V(1,1)$ cycle inverting at the coarsest grid level. Tables 5.1 – 5.20 summarize the results for this problems class using these *unity-based* kink-factors.

Following these results, we present spectral radii results using eigenvector-based kink-factors. Since this calculation is based upon the spatial shape of the slowest converging error mode, only results from the Fourier analysis code are presented.(Recall that no purely-scattering $(c = 1.0)$ are obtainable for the infinite medium problem; thus, we do not present Fourier-analysis results for purely-scattering problems.) While it may be possible to use similar kink-factors in the general solver, we present these results as an indication that it may not make enough of a difference to justify the effort. The concept of designing coarse-grid operators to eliminate a specific error mode is an attractive theoretic premise; however, the use of a unit, isotropic boundary condition is attractive from the viewpoints of simplicity and computational overhead.

Table 5.1 – Predicted and Observed Convergence Ratios, c = 1.0, LBLD

$\sigma_t h_x = \sigma_t h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	N/A	0.472	0.464	0.463	0.458
1.0×10^{-3}	N/A	0.547	0.526	0.519	0.504
1.0×10^{-2}	N/A	0.617	0.512	0.500	0.500
1.0×10^{-1}	N/A	0.348	0.278	0.310	0.352
1.0×10^{0}	N/A	0.590	0.566	0.559	0.558
1.0×10^{1}	N/A	0.943	0.937	0.935	0.933
1.0×10^{2}	N/A	0.990	0.990	0.988	0.981
1.0×10^{3}	N/A	0.997	0.997	0.995	0.991
1.0×10^{4}	N/A	0.9993	0.9993	0.993	0.974

Table 5.2 – Predicted and Observed Convergence Ratios, c = 1.0, BLD

$\sigma_t h_x = \sigma_t h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	N/A	0.649	0.639	0.638	0.635
1.0×10^{-3}	N/A	0.724	0.703	0.697	0.687
1.0×10^{-2}	N/A	0.782	0.703	0.688	0.673
1.0×10^{-1}	N/A	0.563	0.479	0.435	0.461
1.0×10^{0}	N/A	0.589	0.565	0.558	0.556
1.0×10^{1}	N/A	0.942	0.937	0.935	0.933
1.0×10^{2}	N/A	0.990	0.990	0.988	0.981
1.0×10^{3}	N/A	0.996	0.996	0.994	0.991
1.0×10^{4}	N/A	0.9995	0.9995	0.995	0.984

Table 5.3 – Predicted and Observed Convergence Ratios, c = 0.999999, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.868	0.472	0.464	0.463	0.458
1.0×10^{-3}	0.825	0.547	0.526	0.519	0.504
1.0×10^{-2}	0.701	0.617	0.512	0.500	0.500
1.0×10^{-1}	0.355	0.348	0.278	0.310	0.352
1.0×10^{0}	0.179	0.590	0.566	0.561	0.558
1.0×10^{1}	0.760	0.941	0.935	0.933	0.931
1.0×10^{2}	0.971	0.981	0.980	0.978	0.975
1.0×10^{3}	0.993	0.991	0.990	0.989	0.987
1.0×10^{4}	0.977	0.970	0.969	0.967	0.953

Table 5.4 – Predicted and Observed Convergence Ratios, c = 0.999999, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.976	0.649	0.639	0.638	0.635
1.0×10^{-3}	0.958	0.724	0.703	0.697	0.687
1.0×10^{-2}	0.891	0.782	0.703	0.688	0.688
1.0×10^{-1}	0.578	0.563	0.479	0.435	0.461
1.0×10^{0}	0.164	0.589	0.565	0.557	0.561
1.0×10^{1}	0.741	0.940	0.935	0.933	0.931
1.0×10^{2}	0.969	0.981	0.980	0.978	0.975
1.0×10^{3}	0.993	0.991	0.991	0.990	0.989
1.0×10^{4}	0.989	0.981	0.979	0.977	0.973

Table 5.5 – Predicted and Observed Convergence Ratios, c = 0.9999, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.868	0.472	0.464	0.463	0.458
1.0×10^{-3}	0.825	0.547	0.526	0.519	0.504
1.0×10^{-2}	0.701	0.617	0.512	0.500	0.500
1.0×10^{-1}	0.355	0.348	0.278	0.310	0.352
1.0×10^{0}	0.179	0.578	0.552	0.545	0.542
1.0×10^{1}	0.753	0.850	0.835	0.831	0.829
1.0×10^{2}	0.934	0.930	0.924	0.922	0.921
1.0×10^{3}	0.801	0.797	0.780	0.774	0.771
1.0×10^{4}	0.222	0.221	0.199	0.193	0.190

Table 5.6 – Predicted and Observed Convergence Ratios, c = 0.9999, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.976	0.649	0.639	0.638	0.635
1.0×10^{-3}	0.958	0.724	0.703	0.697	0.686
1.0×10^{-2}	0.891	0.782	0.703	0.688	0.673
1.0×10^{-1}	0.578	0.563	0.479	0.435	0.461
1.0×10^{0}	0.164	0.578	0.551	0.543	0.540
1.0×10^{1}	0.734	0.849	0.834	0.830	0.828
1.0×10^{2}	0.936	0.932	0.926	0.924	0.923
1.0×10^{3}	0.902	0.896	0.887	0.884	0.882
1.0×10^{4}	0.451	0.449	0.420	0.412	0.407

Table 5.7 – Predicted and Observed Convergence Ratios, c = 0.99, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.867	0.472	0.464	0.462	0.457
1.0×10^{-3}	0.824	0.547	0.526	0.518	0.504
1.0×10^{-2}	0.700	0.615	0.511	0.500	0.500
1.0×10^{-1}	0.353	0.346	0.277	0.309	0.352
1.0×10^{0}	0.174	0.234	0.196	0.189	0.207
1.0×10^{1}	0.531	0.530	0.504	0.495	0.491
1.0×10^{2}	0.213	0.213	0.191	0.185	0.183
1.0×10^{3}	6.86×10^{-3}	6.85×10^{-3}	5.70×10^{-3}	5.41×10^{-3}	5.31×10^{-3}
1.0×10^{4}	8.09×10^{-5}	7.99×10^{-5}	6.57×10^{-5}	6.21×10^{-5}	6.08×10^{-5}

Table 5.8 – Predicted and Observed Convergence Ratios, c = 0.99, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.976	0.648	0.638	0.638	0.635
1.0×10^{-3}	0.957	0.723	0.703	0.696	0.686
1.0×10^{-2}	0.890	0.781	0.702	0.688	0.672
1.0×10^{-1}	0.576	0.561	0.478	0.435	0.460
1.0×10^{0}	0.160	0.232	0.239	0.341	0.384
1.0×10^{1}	0.551	0.549	0.519	0.510	0.506
1.0×10^{2}	0.424	0.423	0.394	0.386	0.382
1.0×10^{3}	2.92×10^{-2}	2.91×10^{-2}	2.46×10^{-2}	2.35×10^{-2}	2.31×10^{-2}
1.0×10^{4}	4.10×10^{-4}	4.09×10^{-4}	3.37×10^{-4}	3.18×10^{-4}	3.12×10^{-4}

Table 5.9 – Predicted and Observed Convergence Ratios, c = 0.9, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.866	0.469	0.462	0.460	0.455
1.0×10^{-3}	0.822	0.542	0.522	0.515	0.501
1.0×10^{-2}	0.693	0.604	0.506	0.496	0.500
1.0×10^{-1}	0.335	0.327	0.272	0.306	0.349
1.0×10^{0}	0.133	0.133	0.128	0.125	0.124
1.0×10^{1}	0.145	0.145	0.127	0.121	0.119
1.0×10^{2}	6.08×10^{-3}	6.07×10^{-3}	5.01×10^{-3}	4.74×10^{-3}	4.65×10^{-3}
1.0×10^{3}	7.29×10^{-5}	7.28×10^{-5}	5.94×10^{-5}	5.60×10^{-5}	5.49×10^{-5}
1.0×10^{4}	7.42×10^{-7}	7.42×10^{-7}	6.04×10^{-7}	5.70×10^{-7}	5.58×10^{-7}

Table 5.10 – Predicted and Observed Convergence Ratios, c = 0.9, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.975	0.646	0.636	0.636	0.632
1.0×10^{-3}	0.956	0.719	0.699	0.693	0.683
1.0×10^{-2}	0.886	0.774	0.698	0.684	0.669
1.0×10^{-1}	0.556	0.538	0.484	0.433	0.454
1.0×10^{0}	0.140	0.134	0.116	9.71×10^{-2}	0.109
1.0×10^{1}	0.244	0.244	0.217	0.209	0.206
1.0×10^{2}	2.54×10^{-2}	2.54×10^{-2}	2.13×10^{-2}	2.02×10^{-2}	1.98×10^{-2}
1.0×10^{3}	3.64×10^{-4}	3.64×10^{-4}	2.97×10^{-4}	2.80×10^{-4}	2.74×10^{-4}
1.0×10^{4}	3.78×10^{-6}	3.78×10^{-6}	3.07×10^{-6}	2.90×10^{-6}	2.84×10^{-6}

Table 5.11 – Predicted and Observed Convergence Ratios, c = 0.7, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.865	0.463	0.455	0.454	0.449
1.0×10^{-3}	0.816	0.531	0.513	0.506	0.493
1.0×10^{-2}	0.676	0.579	0.493	0.486	0.528
1.0×10^{-1}	0.298	0.290	0.261	0.298	0.342
1.0×10^{0}	7.76×10^{-2}	7.71×10^{-2}	6.74×10^{-2}	6.48×10^{-2}	6.44×10^{-2}
1.0×10^{1}	2.80×10^{-2}	2.80×10^{-2}	2.30×10^{-2}	2.17×10^{-2}	2.12×10^{-2}
1.0×10^{2}	5.81×10^{-4}	5.80×10^{-4}	4.67×10^{-4}	4.40×10^{-4}	4.30×10^{-4}
1.0×10^{3}	6.26×10^{-6}	6.25×10^{-6}	5.02×10^{-6}	4.72×10^{-6}	4.62×10^{-6}
1.0×10^{4}	6.30×10^{-8}	6.30×10^{-8}	5.06×10^{-8}	4.76×10^{-8}	4.65×10^{-8}

Table 5.12 – Predicted and Observed Convergence Ratios, c = 0.7, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.974	0.638	0.629	0.629	0.625
1.0×10^{-3}	0.954	0.709	0.690	0.685	0.676
1.0×10^{-2}	0.877	0.756	0.686	0.674	0.662
1.0×10^{-1}	0.517	0.501	0.472	0.428	0.449
1.0×10^{0}	0.107	0.105	8.13×10^{-2}	5.29×10^{-2}	6.58×10^{-2}
1.0×10^{1}	7.42×10^{-2}	7.41×10^{-2}	6.15×10^{-2}	5.80×10^{-2}	5.68×10^{-2}
1.0×10^{2}	2.60×10^{-3}	2.60×10^{-3}	2.09×10^{-3}	1.96×10^{-3}	1.92×10^{-3}
1.0×10^{3}	3.03×10^{-5}	2.94×10^{-5}	2.35×10^{-5}	2.20×10^{-5}	2.15×10^{-5}
1.0×10^{4}	3.09×10^{-7}	2.97×10^{-7}	2.37×10^{-7}	2.23×10^{-7}	2.17×10^{-7}

Table 5.13 – Predicted and Observed Convergence Ratios, c = 0.5, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.863	0.454	0.447	0.446	0.441
1.0×10^{-3}	0.811	0.517	0.501	0.495	0.483
1.0×10^{-2}	0.662	0.553	0.478	0.474	0.500
1.0×10^{-1}	0.266	0.256	0.249	0.291	0.334
1.0×10^{0}	4.67×10^{-2}	4.64×10^{-2}	3.65×10^{-2}	3.55×10^{-2}	3.52×10^{-2}
1.0×10^{1}	8.73×10^{-3}	8.72×10^{-3}	6.96×10^{-3}	6.50×10^{-3}	6.34×10^{-3}
1.0×10^{2}	1.49×10^{-4}	1.49×10^{-4}	1.19×10^{-4}	1.11×10^{-4}	1.09×10^{-4}
1.0×10^{3}	1.58×10^{-6}	1.57×10^{-6}	1.25×10^{-6}	1.17×10^{-6}	1.15×10^{-6}
1.0×10^{4}	1.58×10^{-8}	1.58×10^{-8}	1.26×10^{-8}	1.18×10^{-8}	1.15×10^{-8}

Table 5.14 – Predicted and Observed Convergence Ratios, c = 0.5, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.974	0.628	0.620	0.620	0.617
1.0×10^{-3}	0.951	0.697	0.679	0.674	0.666
1.0×10^{-2}	0.868	0.736	0.671	0.663	0.651
1.0×10^{-1}	0.483	0.464	0.459	0.423	0.438
1.0×10^{0}	8.27×10^{-2}	8.18×10^{-2}	6.61×10^{-2}	4.24×10^{-2}	5.95×10^{-2}
1.0×10^{1}	2.62×10^{-2}	2.62×10^{-2}	2.07×10^{-2}	1.96×10^{-2}	1.93×10^{-2}
1.0×10^{2}	7.05×10^{-4}	6.92×10^{-4}	5.54×10^{-4}	5.24×10^{-4}	5.14×10^{-4}
1.0×10^{3}	8.06×10^{-6}	7.84×10^{-6}	6.23×10^{-6}	5.90×10^{-6}	5.79×10^{-6}
1.0×10^{4}	8.18×10^{-8}	7.84×10^{-8}	6.24×10^{-8}	5.91×10^{-8}	5.80×10^{-8}

Table 5.15 – Predicted and Observed Convergence Ratios, c = 0.3, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.861	0.442	0.436	0.434	0.430
1.0×10^{-3}	0.807	0.499	0.485	0.480	0.469
1.0×10^{-2}	0.648	0.522	0.460	0.458	0.480
1.0×10^{-1}	0.237	0.225	0.236	0.287	0.325
1.0×10^{0}	2.83×10^{-2}	2.80×10^{-2}	1.91×10^{-2}	2.03×10^{-2}	2.07×10^{-2}
1.0×10^{1}	2.89×10^{-3}	2.89×10^{-3}	2.26×10^{-3}	2.10×10^{-3}	2.04×10^{-3}
1.0×10^{2}	4.47×10^{-5}	4.47×10^{-5}	3.53×10^{-5}	3.30×10^{-5}	3.23×10^{-5}
1.0×10^{3}	4.67×10^{-7}	4.66×10^{-7}	3.69×10^{-7}	3.45×10^{-7}	3.37×10^{-7}
1.0×10^{4}	4.69×10^{-9}	4.68×10^{-9}	3.71×10^{-9}	3.47×10^{-9}	3.39×10^{-9}

Table 5.16 – Predicted and Observed Convergence Ratios, c = 0.3, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.973	0.614	0.607	0.608	0.605
1.0×10^{-3}	0.949	0.679	0.663	0.659	0.652
1.0×10^{-2}	0.859	0.709	0.652	0.646	0.638
1.0×10^{-1}	0.453	0.430	0.449	0.416	0.422
1.0×10^{0}	6.18×10^{-2}	6.08×10^{-2}	5.47×10^{-2}	3.66×10^{-2}	5.60×10^{-2}
1.0×10^{1}	1.17×10^{-2}	1.16×10^{-2}	9.53×10^{-3}	9.70×10^{-3}	9.63×10^{-3}
1.0×10^{2}	3.16×10^{-4}	2.85×10^{-4}	2.50×10^{-4}	2.63×10^{-4}	2.62×10^{-4}
1.0×10^{3}	3.85×10^{-6}	3.82×10^{-6}	3.25×10^{-6}	3.37×10^{-6}	3.51×10^{-6}
1.0×10^{4}	3.97×10^{-8}	3.93×10^{-8}	3.34×10^{-8}	3.46×10^{-8}	3.44×10^{-8}

Table 5.17 – Predicted and Observed Convergence Ratios, c = 0.1, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.859	0.418	0.412	0.410	0.406
1.0×10^{-3}	0.803	0.467	0.455	0.451	0.443
1.0×10^{-2}	0.635	0.476	0.429	0.430	0.452
1.0×10^{-1}	0.212	0.193	0.218	0.254	0.299
1.0×10^{0}	1.82×10^{-2}	1.65×10^{-2}	1.52×10^{-2}	1.58×10^{-2}	1.54×10^{-2}
1.0×10^{1}	1.14×10^{-3}	1.11×10^{-3}	8.71×10^{-4}	9.66×10^{-4}	9.62×10^{-4}
1.0×10^{2}	1.79×10^{-5}	1.70×10^{-5}	1.42×10^{-5}	1.61×10^{-5}	1.62×10^{-5}
1.0×10^{3}	1.87×10^{-7}	1.71×10^{-7}	1.46×10^{-7}	1.63×10^{-7}	1.66×10^{-7}
1.0×10^{4}	1.88×10^{-9}	1.66×10^{-9}	1.39×10^{-9}	1.54×10^{-9}	1.59×10^{-9}

Table 5.18 – Predicted and Observed Convergence Ratios, c = 0.1, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.972	0.587	0.581	0.582	0.581
1.0×10^{-3}	0.947	0.644	0.631	0.628	0.621
1.0×10^{-2}	0.850	0.665	0.617	0.615	0.614
1.0×10^{-1}	0.427	0.400	0.436	0.393	0.404
1.0×10^{0}	4.35×10^{-2}	4.21×10^{-2}	4.65×10^{-2}	3.36×10^{-2}	5.27×10^{-2}
1.0×10^{1}	7.25×10^{-3}	7.15×10^{-3}	6.50×10^{-3}	8.28×10^{-3}	8.25×10^{-3}
1.0×10^{2}	4.86×10^{-4}	4.80×10^{-4}	4.42×10^{-4}	5.47×10^{-4}	5.53×10^{-4}
1.0×10^{3}	8.09×10^{-6}	8.00×10^{-6}	6.99×10^{-6}	8.53×10^{-6}	8.60×10^{-6}
1.0×10^{4}	8.61×10^{-8}	8.51×10^{-8}	7.39×10^{-8}	9.02×10^{-8}	9.10×10^{-8}

Table 5.19 – Predicted and Observed Convergence Ratios, $c = 0.0$, LBLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.859	*a*[†]	*a*	*a*	*a*
1.0×10^{-3}	0.801	*a*	*a*	*a*	*a*
1.0×10^{-2}	0.628	*a*	*a*	*a*	*a*
1.0×10^{-1}	0.201	*a*	*a*	*a*	*a*
1.0×10^{0}	1.63×10^{-2}	*a*	*a*	*a*	*a*
1.0×10^{1}	6.47×10^{-4}	5.69×10^{-4}	4.85×10^{-4}	6.97×10^{-4}	7.36×10^{-4}
1.0×10^{2}	1.05×10^{-5}	1.02×10^{-5}	9.27×10^{-6}	1.37×10^{-5}	1.45×10^{-5}
1.0×10^{3}	1.11×10^{-7}	1.08×10^{-7}	9.87×10^{-8}	1.48×10^{-7}	1.55×10^{-7}
1.0×10^{4}	1.11×10^{-9}	1.09×10^{-9}	9.96×10^{-10}	1.49×10^{-9}	1.55×10^{-9}

Table 5.20 – Predicted and Observed Convergence Ratios, $c = 0.0$, BLD

$\sigma h_x = \sigma h_y$	Fourier S_2	S_2	S_4	S_8	S_{16}
1.0×10^{-4}	0.972	*a*	*a*	*a*	*a*
1.0×10^{-3}	0.946	*a*	*a*	*a*	*a*
1.0×10^{-2}	0.845	*a*	*a*	*a*	*a*
1.0×10^{-1}	0.416	*a*	*a*	*a*	*a*
1.0×10^{0}	3.53×10^{-2}	*a*	*a*	*a*	*a*
1.0×10^{1}	6.73×10^{-3}	*a*	*a*	*a*	*a*
1.0×10^{2}	9.98×10^{-4}	*a*	*a*	*a*	*a*
1.0×10^{3}	1.05×10^{-4}	*a*	*a*	*a*	*a*
1.0×10^{4}	1.05×10^{-5}	*a*	*a*	*a*	*a*

[†] See the remarks regarding purely-absorbing problems in the text.

Tables 5.1 – 5.20 show the convergence performance and spectral radii of the two-dimensional multigrid method. While the method is, in general, not as rapidly converging as the one-dimensional method, there are interesting behaviors worth investigating. Consistently, and across the entire problem domain, the Fourier analysis predicts poorer convergence behavior than is observed by S_NAPPER_2D for thin problems. Further, for highly scattering problems, the Fourier analysis predicts better performance for problems with cell sizes of one (1) or ten (10) mean-free-paths. The first discrepancy, poorer performance predicted for thin problems, is a result of the problem run by S_NAPPER_2D being small enough that leakage plays a major role, whereas there is no leakage in the infinite-medium problems of the Fourier analysis. For the purely absorbing problems (marked with an '*a*') in Tables 5.19 and 5.20, simple relaxation on the fine grid is an exact solver following sufficient iterations to propagate information from one side of the problem to another. In the case of a V(1,1) cycle, this corresponds to 64 iterations for our 128×128 test cases. As a result, in our finite test problems the method sometimes converges before the error ratio asymptotically approaches a value that can be interpreted as the spectral radius of the iteration operator.

Tables 5.21 and 5.22 reflect when our Fourier analysis is a reliably predictive measure of the convergence performance of the two-dimensional method. For thin problems, our Fourier analysis is providing an upper bound for the convergence ratio. For problems as large are we are able to run (*i.e.*, 512×512, S_{16}) the observed convergence ratio of S_NAPPER_2D approaches the calculated spectral radius from below. This is not the case for the intermediate problems of one and ten mean-free-paths. In these cases, we are seeing the difference between inverting on grid level 16ℓ, as the Fourier analysis does, and continuing to coarsen to levels further down the V-cycle, as S_NAPPER_2D does. Error modes that exist on coarser levels are represented by the Fourier analysis; however, they are being eliminated by the direct inversion step. This also provides insight into the thick-problem behavior. For thick problems, as we saw in one dimension, if performance degrades it is due to inaccurately characterizing the error on the finest grid. Since this effect is experienced by both the Fourier analysis codes and

S_NAPPER_2D on the finest grid level, the two are in excellent agreement across a wide range of problem sizes.

Table 5.21 – Convergence Ratios by Problem Size, *c* = 0.999999, LBLD

$\sigma h_x = \sigma h_y$	8×8	32×32	64×64	256×256	512×512
1.0×10^{-4}	4.04×10^{-3}	0.135	0.303	0.633	0.778
1.0×10^{-3}	9.54×10^{-3}	0.192	0.379	0.713	0.828
1.0×10^{-2}	2.40×10^{-2}	0.279	0.464	0.730	0.784
1.0×10^{-1}	5.34×10^{-2}	0.254	0.322	0.355	0.355
1.0×10^{0}	0.147	0.210	0.500	0.750	0.856
1.0×10^{1}	0.690	0.829	0.897	0.963	0.974
1.0×10^{2}	0.960	0.976	0.980	0.980	0.982
1.0×10^{3}	0.991	0.991	0.991	0.991	0.991
1.0×10^{4}	0.961	0.969	0.970	0.971	0.971

Table 5.22 – Convergence Ratios by Problem Size, c = 0.999999, BLD

$\sigma h_x = \sigma h_y$	8×8	32×32	64×64	256×256	512×512
1.0×10^{-4}	1.38×10^{-2}	0.269	0.460	0.789	0.870
1.0×10^{-3}	2.63×10^{-2}	0.342	0.543	0.837	0.897
1.0×10^{-2}	6.02×10^{-2}	0.443	0.642	0.849	0.877
1.0×10^{-1}	0.126	0.451	0.530	0.574	0.575
1.0×10^{0}	0.124	0.206	0.499	0.749	0.856
1.0×10^{1}	0.633	0.827	0.897	0.964	0.974
1.0×10^{2}	0.951	0.976	0.980	0.982	0.982
1.0×10^{3}	0.992	0.992	0.991	0.991	0.991
1.0×10^{4}	0.978	0.980	0.980	0.980	0.980

Tables 5.23 – 5.26 present spectral radii for the same set of problems (with the exception of purely-scattering) using the slowest-converging error mode to compute kink-factors. We observe convergence behavior that is more rapid in the thick limit; however, there is no analogous improvement in the thin and intermediate limit (keeping in mind the shortcoming of a three-level Fourier analysis previous discussed).

For problems dominated by scattering, this two-dimensional multigrid method represents an improvement over simple source iteration (Tables 5.23 and 5.24). For problem that are thin and dominated by absorption, we see poor convergence performance using either method for computing kink-factors (Tables 5.25 and 5.26). Further analysis of the thin limit shows a slowest converging error mode that is discontinuous everywhere. As discussed in Chapters II and IV, if we try to make our kink-factors more realistic (discontinuous or anisotropic), we significantly increase the computational cost of the method. In one dimension, difficult problems forced a re-examination of this multigrid method as a stand-alone solver. In two dimensions, we see

degradation even for simple problems. This strengthens the argument for using this multigrid method as a preconditioner.

Table 5.23 – Spectral Radii Using Eigenvector-Based Kink-Factors, $c \geq 0.9$, LBLD

$\sigma h_x = \sigma h_y$	$c = 0.999999$	$c = 0.9999$	$c = 0.99$	$c = 0.9$
1.0×10^{-4}	0.891	0.892	0.875	0.875
1.0×10^{-3}	0.858	0.868	0.857	0.878
1.0×10^{-2}	0.714	0.720	0.733	0.701
1.0×10^{-1}	0.355	0.355	0.353	0.334
1.0×10^{0}	0.179	0.179	0.174	0.133
1.0×10^{1}	0.760	0.753	0.522	9.71×10^{-2}
1.0×10^{2}	0.971	0.930	0.130	2.50×10^{-3}
1.0×10^{3}	0.993	0.724	2.93×10^{-3}	2.68×10^{-5}
1.0×10^{4}	0.966	0.134	3.23×10^{-5}	2.70×10^{-7}

Table 5.24 – Spectral Radii Using Eigenvector-Based Kink-Factors, $c \geq 0.9$, BLD

$\sigma h_x = \sigma h_y$	$c = 0.999999$	$c = 0.9999$	$c = 0.99$	$c = 0.9$
1.0×10^{-4}	0.980	0.981	0.980	0.984
1.0×10^{-3}	0.960	0.961	0.963	0.957
1.0×10^{-2}	0.891	0.891	0.890	0.886
1.0×10^{-1}	0.579	0.584	0.583	0.561
1.0×10^{0}	0.164	0.164	0.160	0.140
1.0×10^{1}	0.741	0.734	0.550	0.227
1.0×10^{2}	0.969	0.934	0.428	2.82×10^{-2}
1.0×10^{3}	0.993	0.898	3.16×10^{-2}	4.17×10^{-4}
1.0×10^{4}	0.989	0.457	4.70×10^{-4}	4.35×10^{-6}

Table 5.25 – Spectral Radii Using Eigenvector-Based Kink-Factors, $c \leq 0.7$, LBLD

$\sigma h_x = \sigma h_y$	$c = 0.7$	$c = 0.5$	$c = 0.3$	$c = 0.1$	$c = 0.0$
1.0×10^{-4}	0.876	0.874	0.880	0.874	0.892
1.0×10^{-3}	0.852	0.851	0.878	0.867	0.841
1.0×10^{-2}	0.681	0.666	0.651	0.656	0.629
1.0×10^{-1}	0.297	0.264	0.235	0.210	0.198
1.0×10^{0}	7.69×10^{-2}	4.62×10^{-2}	2.80×10^{-2}	1.86×10^{-2}	1.70×10^{-2}
1.0×10^{1}	1.55×10^{-2}	4.58×10^{-3}	1.57×10^{-3}	4.38×10^{-4}	5.66×10^{-5}
1.0×10^{2}	1.89×10^{-4}	3.91×10^{-5}	9.30×10^{-6}	1.64×10^{-6}	2.09×10^{-8}
1.0×10^{3}	1.82×10^{-6}	3.37×10^{-7}	6.32×10^{-8}	5.09×10^{-9}	1.70×10^{-11}
1.0×10^{4}	1.82×10^{-8}	3.34×10^{-9}	6.14×10^{-10}	4.22×10^{-11}	6.55×10^{-13}

Table 5.26 – Spectral Radii Using Eigenvector-Based Kink-Factors, $c \leq 0.7$, BLD

$\sigma h_x = \sigma h_y$	$c = 0.7$	$c = 0.5$	$c = 0.3$	$c = 0.1$	$c = 0.0$
1.0×10^{-4}	0.979	0.980	0.981	0.979	0.978
1.0×10^{-3}	0.954	0.952	0.949	0.947	0.946
1.0×10^{-2}	0.877	0.868	0.858	0.849	0.845
1.0×10^{-1}	0.524	0.496	0.458	0.429	0.413
1.0×10^{0}	0.107	8.21×10^{-2}	6.13×10^{-2}	5.29×10^{-2}	4.22×10^{-2}
1.0×10^{1}	8.08×10^{-2}	3.27×10^{-2}	1.29×10^{-2}	4.84×10^{-3}	3.76×10^{-3}
1.0×10^{2}	3.22×10^{-3}	8.73×10^{-4}	2.67×10^{-4}	9.93×10^{-5}	6.22×10^{-5}
1.0×10^{3}	3.73×10^{-5}	9.55×10^{-6}	2.95×10^{-6}	1.07×10^{-6}	6.62×10^{-7}
1.0×10^{4}	3.79×10^{-7}	9.64×10^{-8}	2.98×10^{-8}	1.08×10^{-8}	6.66×10^{-9}

Two-Dimensional Numerical Results – Homogeneous Material and Non-Uniform Grid Problems

This class of problems is examined with the same material, discretization, and quadrature properties as the homogeneous, uniform-grid problems. The partitioning of the spatial problem, however is no longer uniform. We use a two-dimensional analog of the one-dimensional test suite for this class of problems. More specifically, we construct a problem where each cell is an order of magnitude greater than its neighbor to the left where the left-most cell is a single mean-free-path thick (Figure 5.5). For each scattering ratio in our problem domain, sixty-four problems are tested. Each problem represents the single mean-free-path horizontal and vertical strips of eight cells being moved from top-to-bottom, and left-to-right respectively. Spectral radius and convergence ratio behavior that is representative of this problem class is given in Table 5.23.

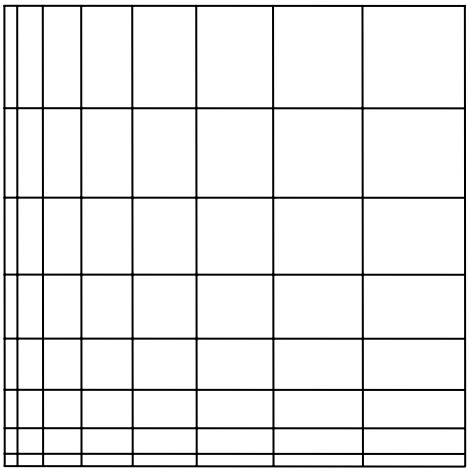

Figure 5.5 – Sixty-Four Cell Repeated Pattern Example for Homogeneous, Non-Uniform Grid Problems

Table 5.27 – Homogeneous, Non-Uniform Test Problems, c = 0.999999; Variation in Thinnest Horizontal Strip Location; LBLD, Kink-Floor OFF

Thinnest Strip Position	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
1	0.980	0.988	0.976
2	0.980	0.988	0.976
3	0.987	0.989	0.986
4	0.987	0.988	0.976
5	0.987	0.987	0.983
6	0.987	0.987	0.981
7	0.987	0.987	0.987
8	0.987	0.987	0.986

Table 5.28 – Homogeneous, Non-Uniform Test Problems, *c* = 0.999999; Variation in Thinnest Horizontal Strip Location; BLD, Kink-Floor OFF

Thinnest Strip Position	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
1	0.990	1.659	1.652
2	0.990	1.657	1.651
3	0.990	1.668	1.661
4	0.990	1.643	1.639
5	0.987	37.501	>>1
6	0.987	2.226	2.216
7	0.987	53.498	51.833
8	0.987	2.227	>>1

Table 5.29 – Homogeneous, Non-Uniform Test Problems, *c* = 0.999999; Variation in Thinnest Vertical Strip Location; LBLD, Kink-Floor OFF

Thinnest Strip Position	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
1	0.980	0.988	0.976
2	0.980	0.988	0.976
3	0.987	0.989	0.986
4	0.987	0.988	0.976
5	0.987	0.987	0.983
6	0.987	0.987	0.981
7	0.987	0.987	0.987
8	0.987	0.987	0.986

Table 5.30 – Homogeneous, Non-Uniform Test Problems, c = 0.999999; Variation in Thinnest Vertical Strip Location; BLD, Kink-Floor OFF

Thinnest Strip Position	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
1	0.990	1.659	1.652
2	0.990	1.657	1.651
3	0.990	1.668	1.661
4	0.990	1.643	1.639
5	0.987	37.501	>>1
6	0.987	2.226	2.216
7	0.987	53.498	51.833
8	0.987	2.227	>>1

In Tables 5.27 – 5.30, we immediately notice the symmetry of the two sets of test problems. For both LBLD and BLD, moving the thinnest horizontal strip from top-to-bottom results in the same convergence behavior as moving the thinnest vertical strip from left-to-right. This is reasonable since, in an infinite medium, the two problems look identical. For LBLD, these problems remained stable; however, this is not the case for BLD (note that in two cases, the convergence ratio never stabilizes to a consistent value – in these instances we record the result as >>1). This should strike the reader as very similar to the behavior observed in the analogous one-dimensional test suite. In Table 5.31, we see that the same instability mitigation tactic of restricting the kink-factors to non-negative values restores stability. The demonstrated symmetry of the problem allows us to demonstrate the benefit of kink-flooring for one set of the homogeneous, non-uniform grid BLD problems.

Table 5.31 – Homogeneous, Non-Uniform Test Problems, *c* = 0.999999; Variation in Thinnest Vertical Strip Location; BLD, Kink-Floor ON

Thinnest Strip Position	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D
1	0.990	0.992	0.982
2	0.990	0.992	0.982
3	0.990	0.991	0.985
4	0.990	0.991	0.986
5	0.987	0.987	0.986
6	0.987	0.987	0.986
7	0.987	0.987	0.986
8	0.987	0.987	0.983

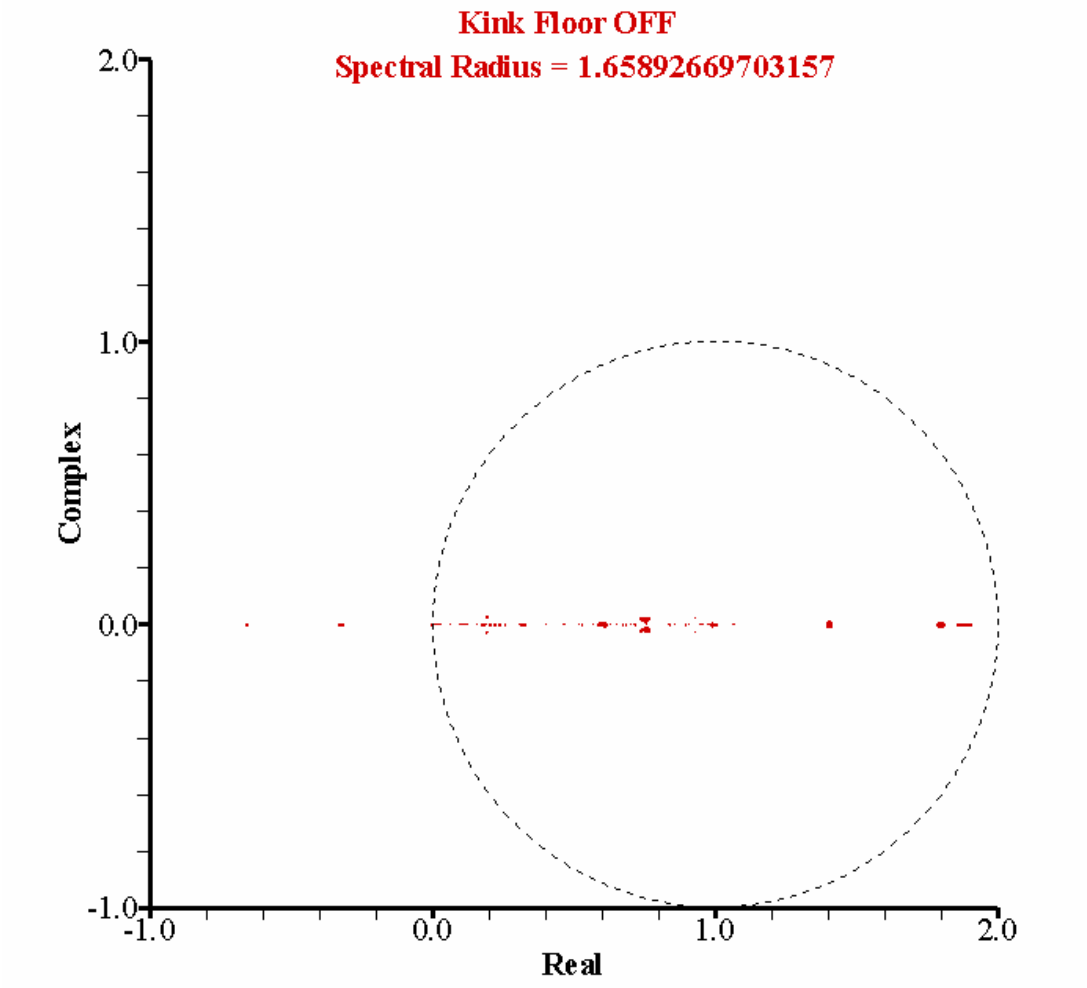

Figure 5.6 – Eigenvalue Map for Homogeneous, Non-Uniform Grid Test Problem 1, BLD, Kink Floor OFF

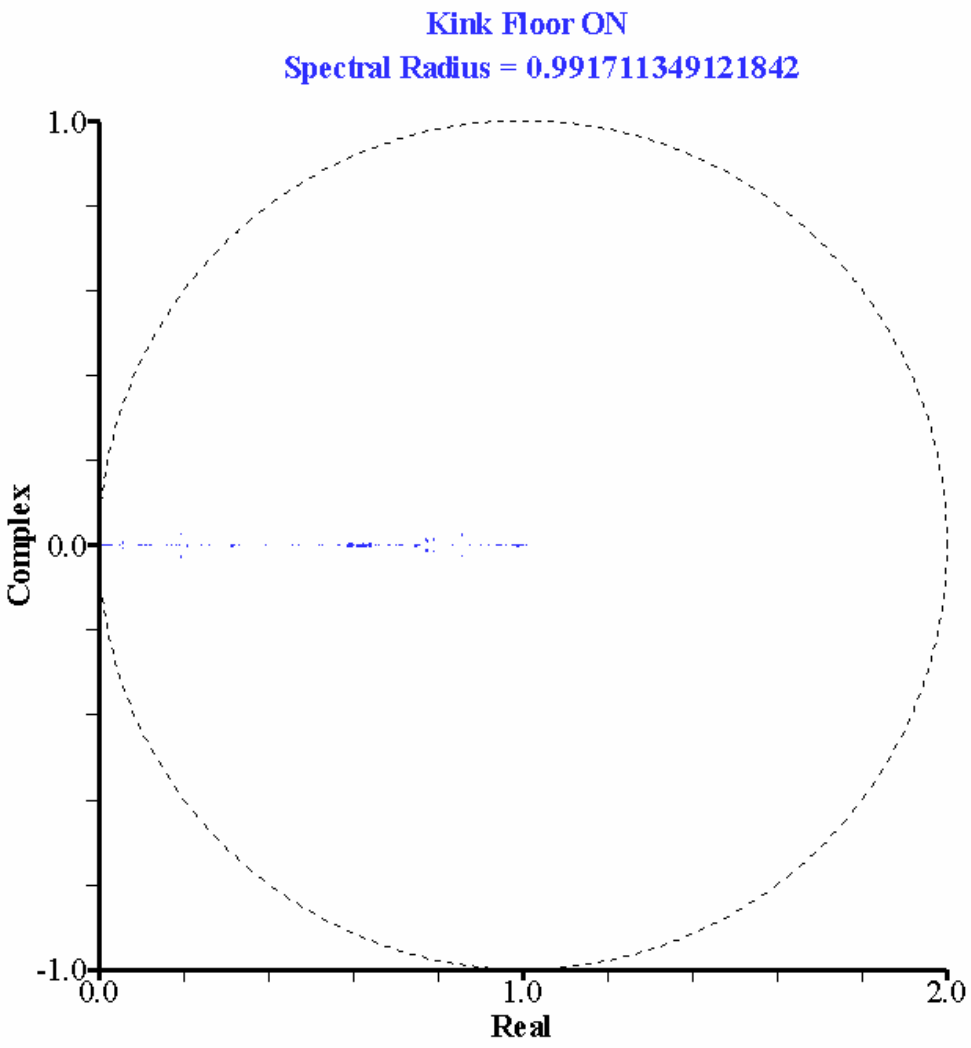

Figure 5.7 – Eigenvalue Map for Homogeneous, Non-Uniform Test Problem 1, BLD, Kink Floor ON

Figures 5.6 and 5.7 show that, even though divergent behavior is observed for this problem using unity-based kink-factors, the eigenvalues corresponding to those divergent modes are very well clustered (Note : there are 716,800 eigenvalues shown in each plot); however, the eigenvalues corresponding to divergent error modes are in the left-half of the real-complex plane. These will not be well attenuated, and may be amplified, by a Krylov solver. As in one dimension, employing a non-negative restriction to the kink-factors eliminates the divergent behavior and does not prevent the desired eigenvalue clustering. Further, it prevents eigenvalues from appearing in the left-half of the real-complex plane. While the spectral radii are comparable for the eigenvector-based and unity-based calculations, it remains an open question which will precondition a Krylov solver more effectively; however, the ease of implementation for the unit-based kink-factor calculation may make it the more desirable option.

Two-Dimensional Numerical Results – Heterogeneous Materials and Uniform Grid Problems

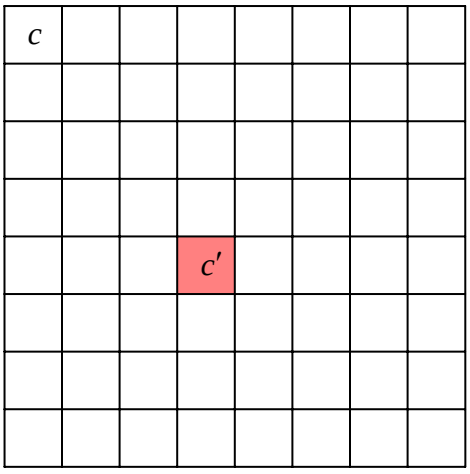

Figure 5.8 – Sixty-Four Cell Repeated Pattern Example for Heterogeneous, Uniform Grid Problems

This class of problems is represented by a uniformly partitioned spatial domain with homogeneous material on every zone except the cell corresponding to cell $K_{(4,4)}$ in the Fourier sixty-four (64) cell pattern (Figure 5.8). A representative problem for this problem class is a highly scattering dominate medium $\left(c = 0.999999\right)$ with a single cell with a slightly smaller scattering ratio $\left(c' = 0.9999\right)$. We choose a problem that performs poorly in the thick limit $\left(\sigma h_x = \sigma h_y = 1.0 \times 10^4\right)$ when using lumping parameters corresponding to BLD $\left(\theta_x = \theta_y = 3.0\right)$. We show results for three distinct methods calculating kink-factors..

Figures 5.9 – 5.11 show results that are distinctly different from any test case in one-dimension. Granted, we have more ways of computing kink-factors in two-dimensions; however, the divergent error modes resulting from negative, unity-based kink-factors have eigenvalues that are outside the unit circle *and* are not well clustered. Another interesting observation is that kink-factors resulting from an eigenvector characterization of the slowest converging error mode do not result in better clustering of eigenvalues than do those generated by a unity-based kink-factor with a non-negative restriction. This complicates but does not discourage this method's suitability as a preconditioner.

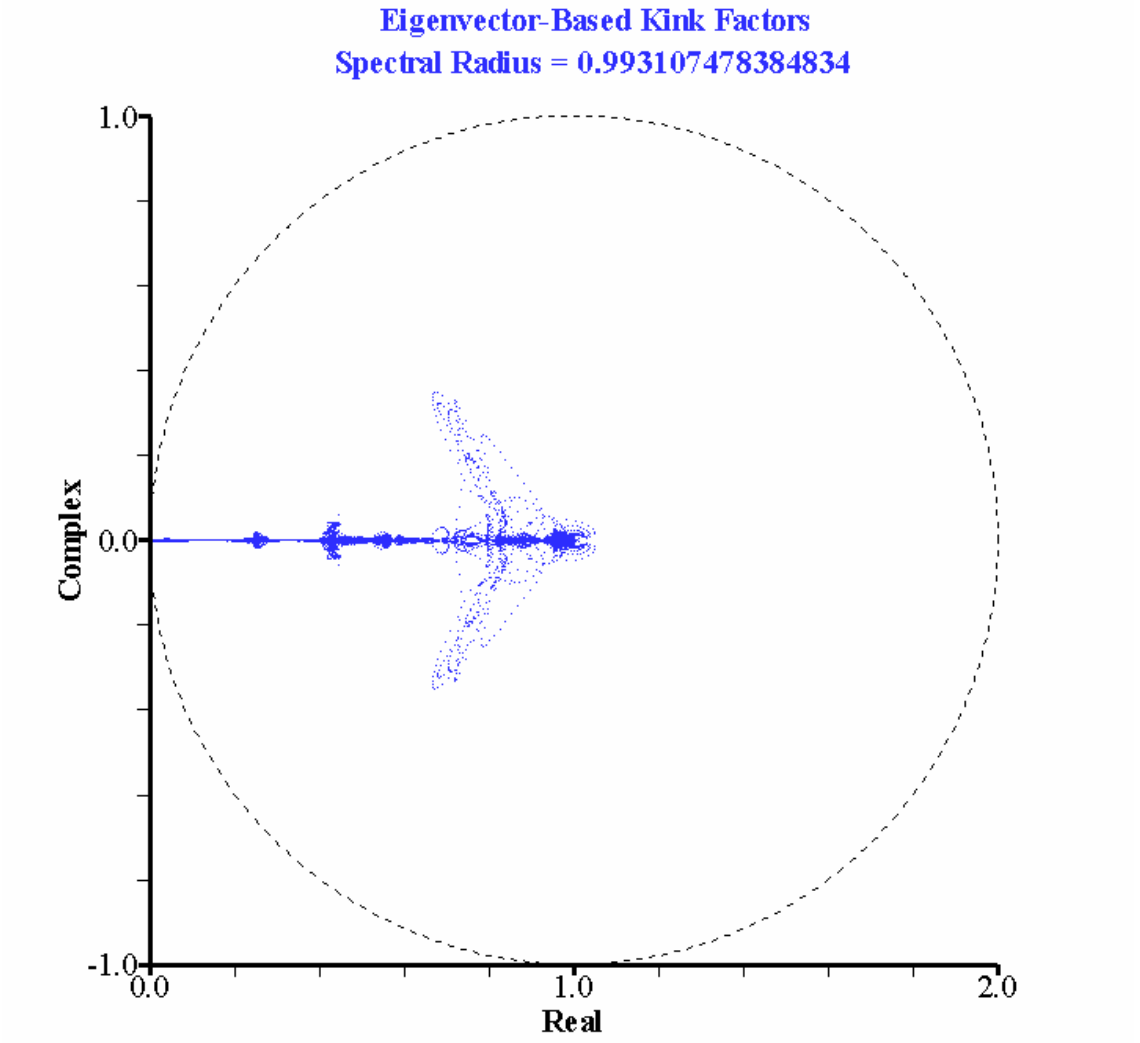

Figure 5.9 – Eigenvalue Map for a Heterogeneous Test Problem, BLD, Eigenvector-Based Kink Factors; Kink Floor OFF

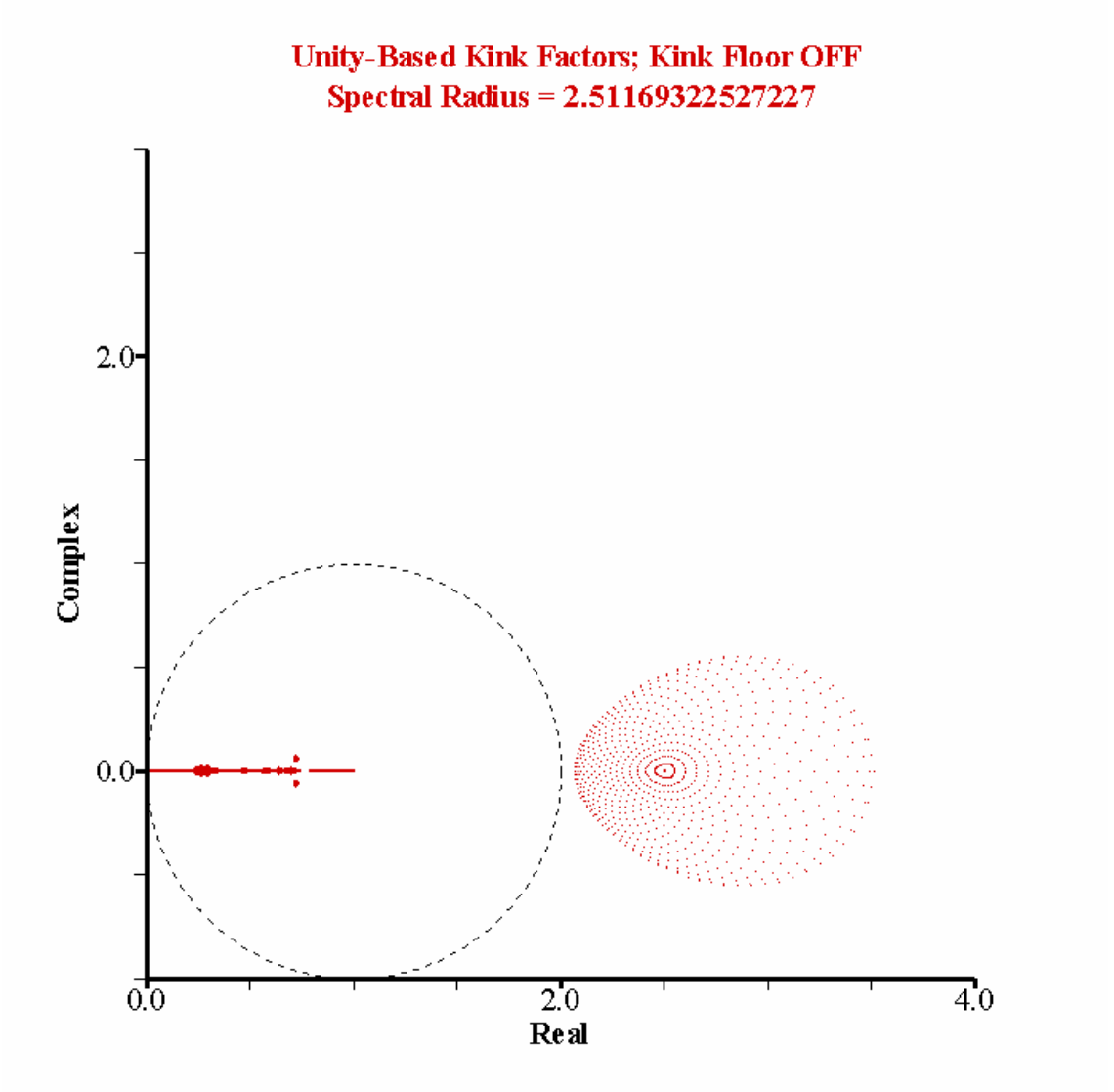

Figure 5.10 – Eigenvalue Map for Heterogeneous Test Problem, BLD, Unity-Based Kink Factors; Kink Floor OFF

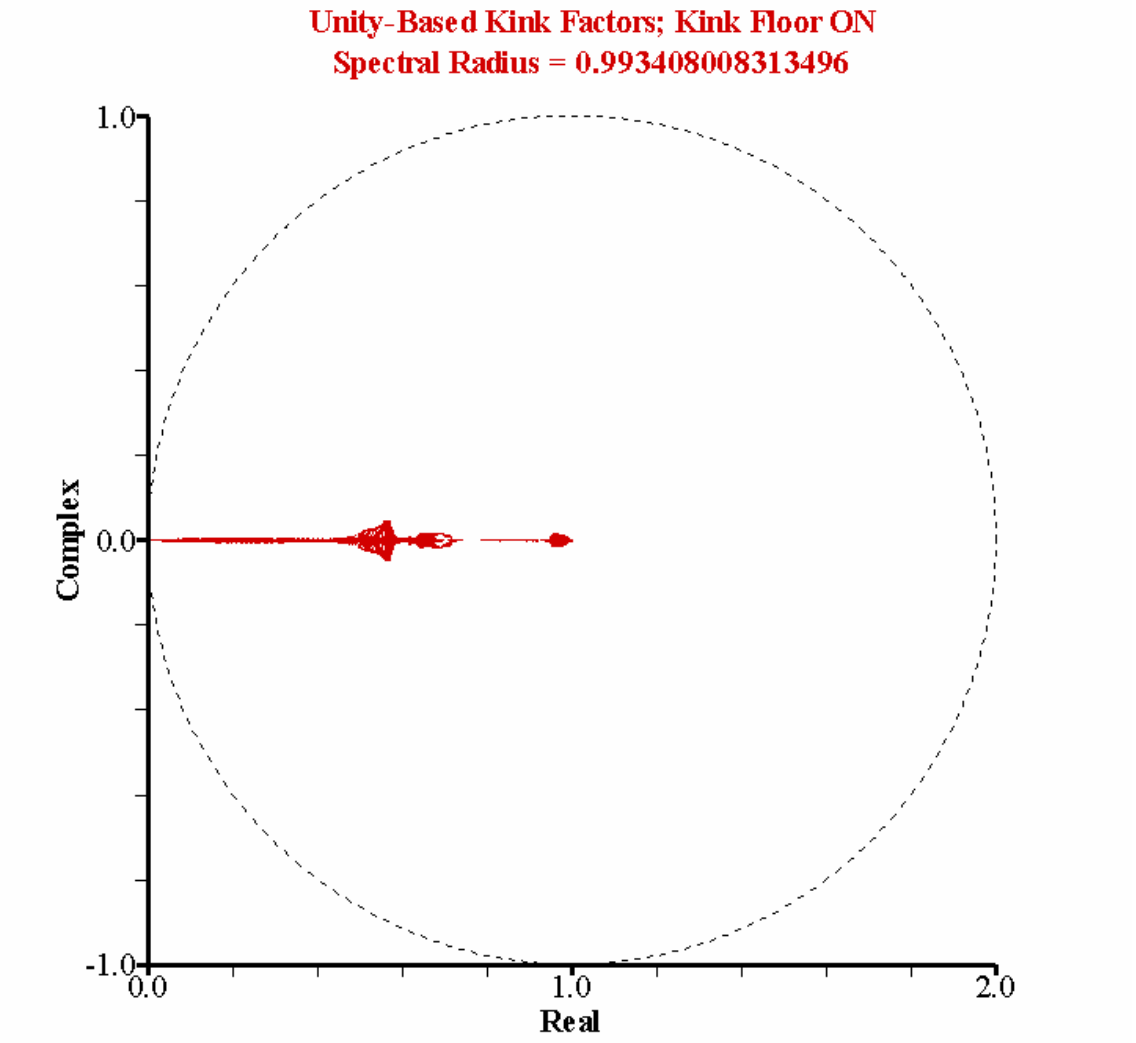

Figure 5.11 – Eigenvalue Map for Heterogeneous Test Problem, BLD, Unity-Based Kink Factors; Kink Floor ON

Two-Dimensional Numerical Results – Heterogeneous Materials and Non-Uniform Grid Problems

The final class of problems for which we show representative results is characterized by heterogeneous material properties partitioned with a non-uniform grid. As in the one-dimensional case, we construct problems that represent a combination of the other three problem classes (Figure 5.12).

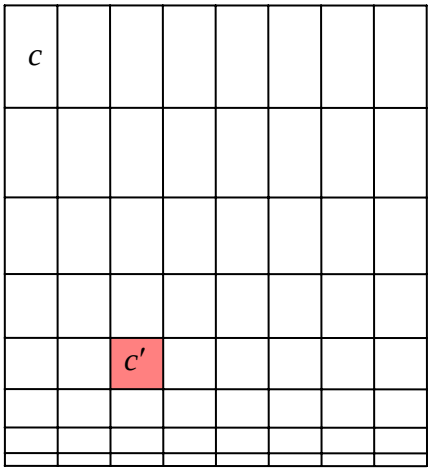

Figure 5.12 – Sixty-Four Cell Repeated Pattern Example for Heterogeneous, Non-Uniform Grid Problems

Table 5.32 – Heterogeneous, Non-Uniform Test Problems; LBLD, Kink-Floor OFF

c'	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
0.9999	0.9993	11.204	1.891
0.99	0.9993	12.593	1.835
0.9	0.9993	13.690	1.832

Table 5.33 – Heterogeneous, Non-Uniform Test Problems; LBLD, Kink-Floor ON

c'	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
0.9999	0.9993	0.9993	0.994
0.99	0.9993	0.9993	0.994
0.9	0.9993	0.9993	0.994

Table 5.34 – Heterogeneous, Non-Uniform Test Problems; BLD, Kink-Floor OFF

c'	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
0.9999	0.9993	1.385	>1
0.99	0.9993	4.383	4.120
0.9	0.9993	1.320	>1

Table 5.35 – Heterogeneous, Non-Uniform Test Problems; BLD, Kink-Floor ON

c'	Fourier S_2 Eigenvector-Based	Fourier S_2 Unity-Based	S_NAPPER_2D Unity-Based
0.9999	0.9993	0.9993	0.995
0.99	0.9993	0.9993	0.995
0.9	0.9993	0.9993	0.995

Tables 5.32 – 5.35 show the results for test cases representing this problem class with $c = 0.999999$. The most significant difference of these results from the other problems classes in the instability observed using the LBLD discretization. This is the first instance where we have observed this behavior using the lumped equations; however, as is shown in every instance of unstable iterative behavior, the restriction of the kink-factors to non-negative values maintains stability. We observe a difference in the predicted (*i.e.*, spectral radius) and the observed convergence behavior; however, the Fourier analysis reliably predicts instability. Quantifying the instability is not our goal. At no point in our analysis do we observe a case where stability or instability is predicted by Fourier analysis and not also observed with the general solver. This numerical-experimentation supports a *sufficient*, but not *necessary*, condition of non-negative kink-factors for stability.

Eigenvalue Analysis

In addition to the Fourier results already presented that represent complete multigrid cycles (*i.e.*, the complete multigrid iteration matrix), we can evaluate the state of the eigenvalue-clustering at discrete stages along the multigrid cycle (recall the comment regarding preconditioned Richardson iteration in Chapter III, Section VII). To demonstrate this, we consider three highly-scattering problems ($c = 0.999999$) representing thin, intermediate and thick cells (0.01, 1.0, and 100.0). For each of these problems, we look at the state of the eigenvalues at four distinct points along the multigrid cycle – 1) one relaxation at grid level ℓ, 2) one relaxation at grid level 4ℓ on the way down the cycle, 3) inversion of the four-cell problem at grid level 16ℓ, and 4) a complete $V(1,1)$ cycle. Figures 5.13 – 5.24 detail this sequence of calculations.

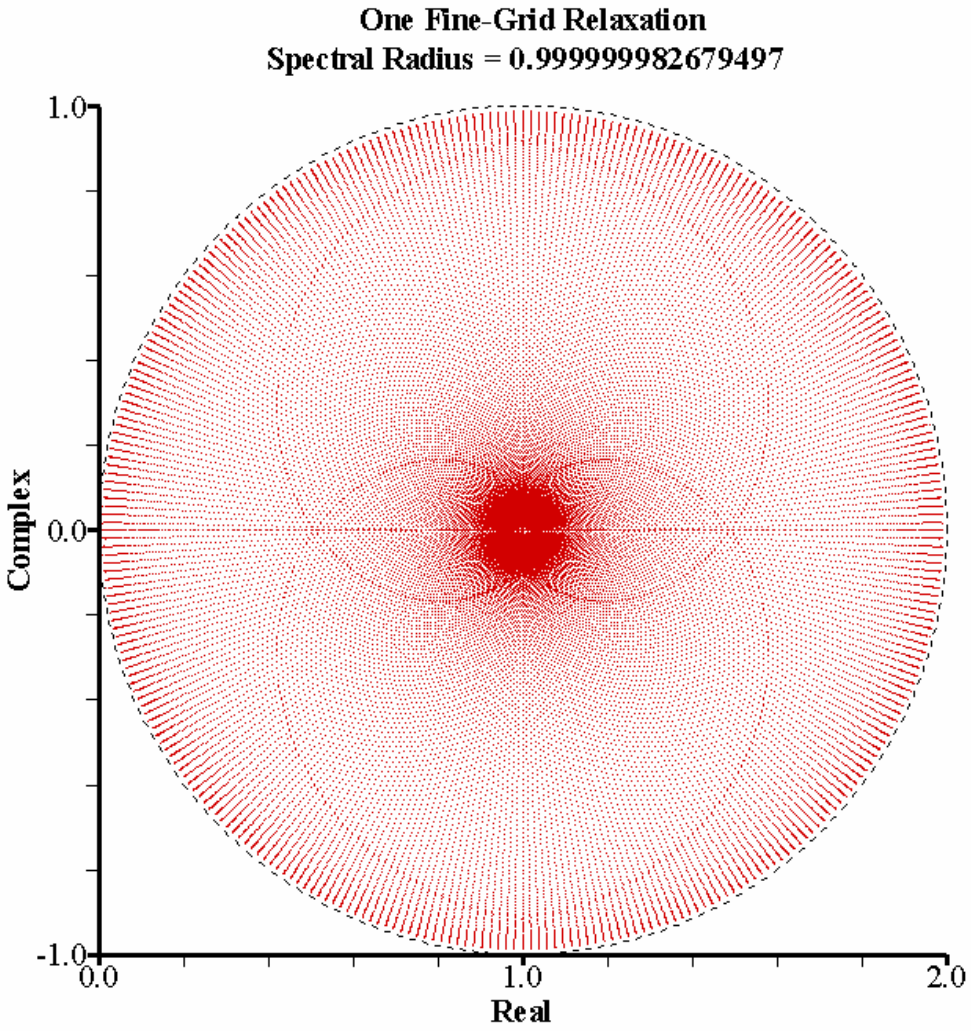

Figure 5.13 – Plot #1 for a Thin Problem, LBLD

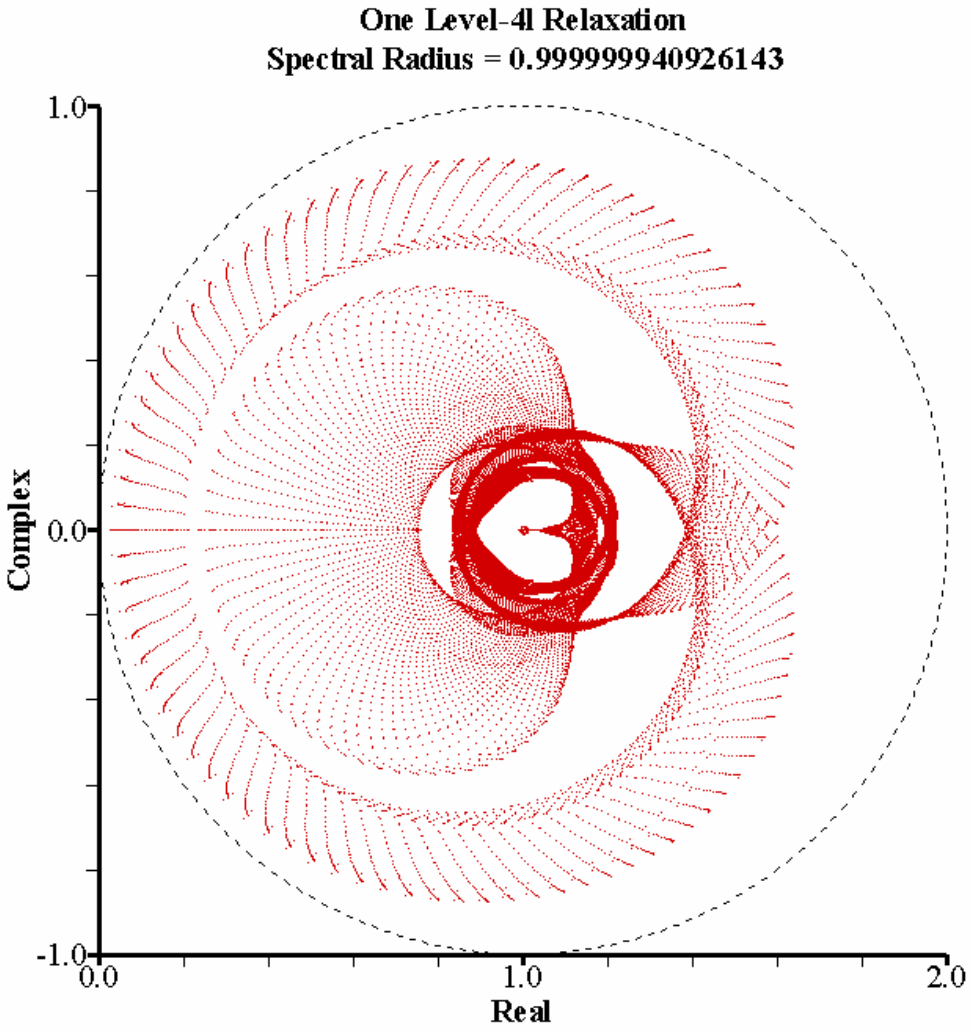

Figure 5.14 – Plot #2 for a Thin Problem, LBLD

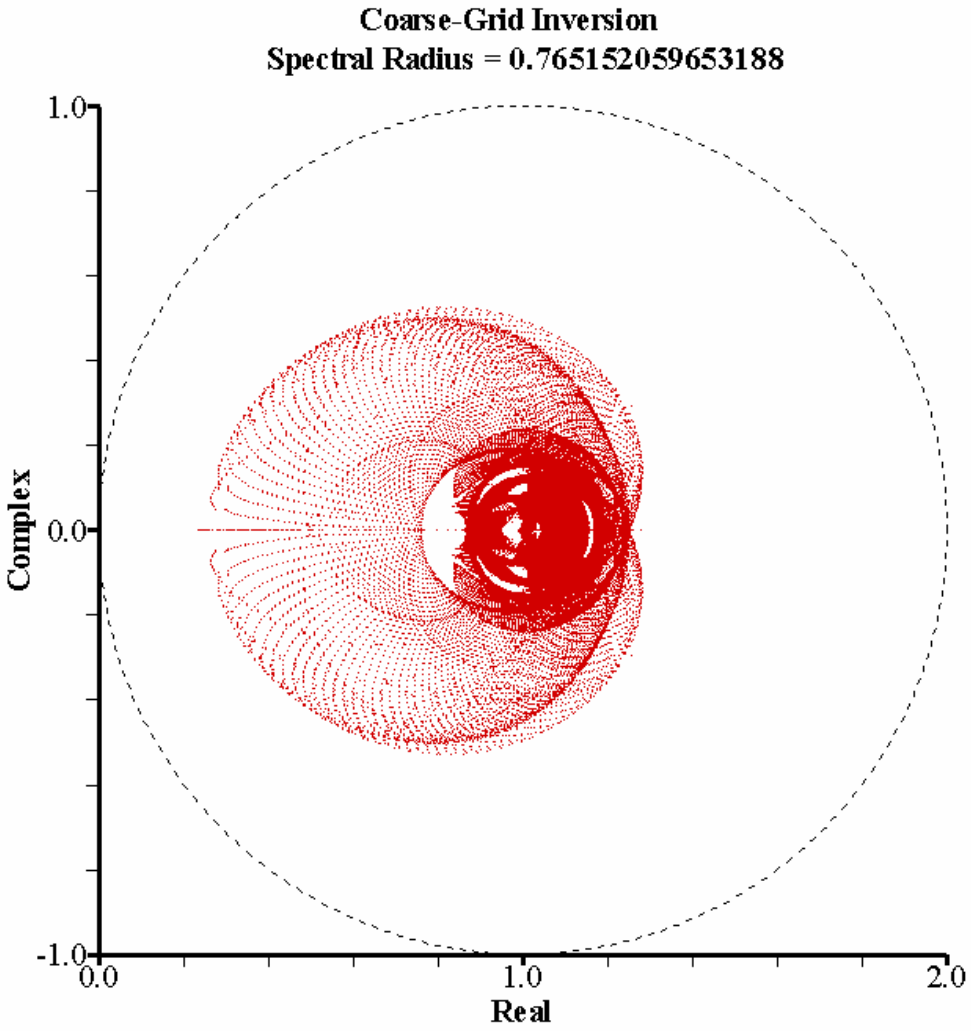

Figure 5.15 – Plot #3 for a Thin Problem, LBLD

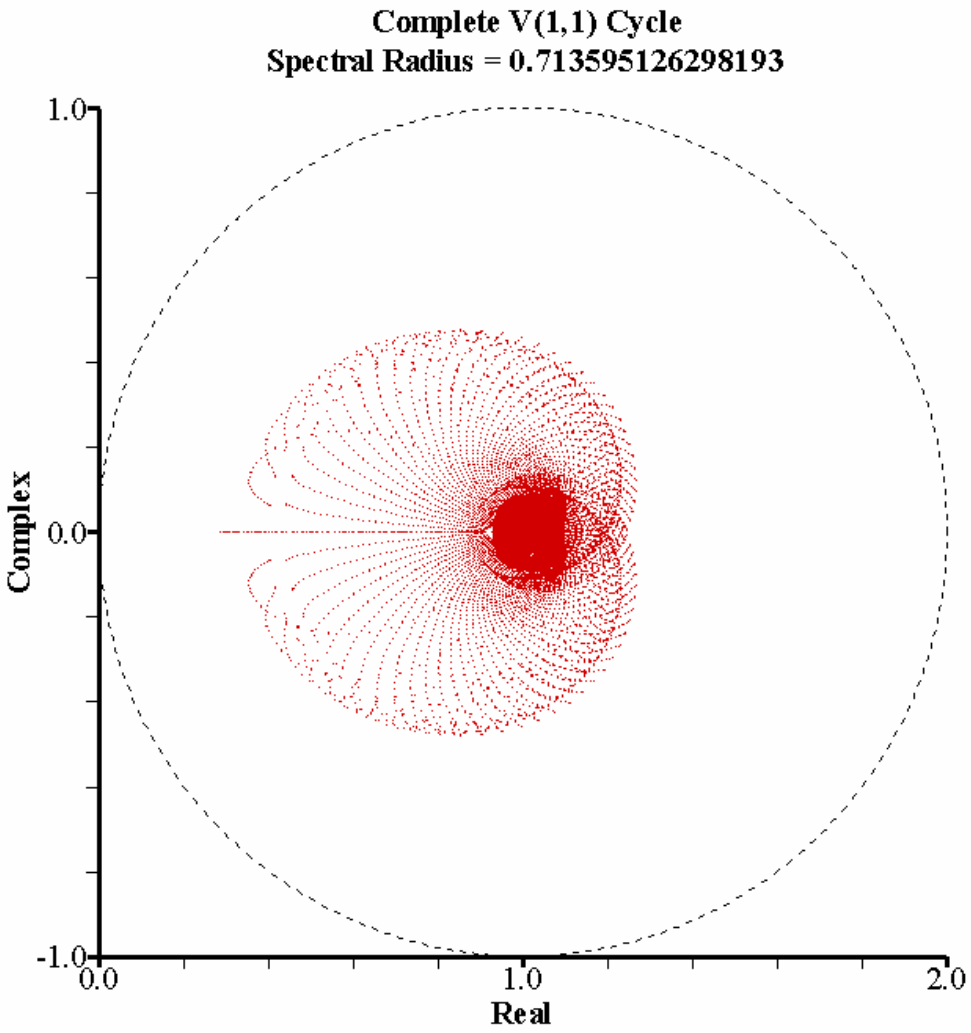

Figure 5.16 – Plot #4 for a Thin Problem, LBLD

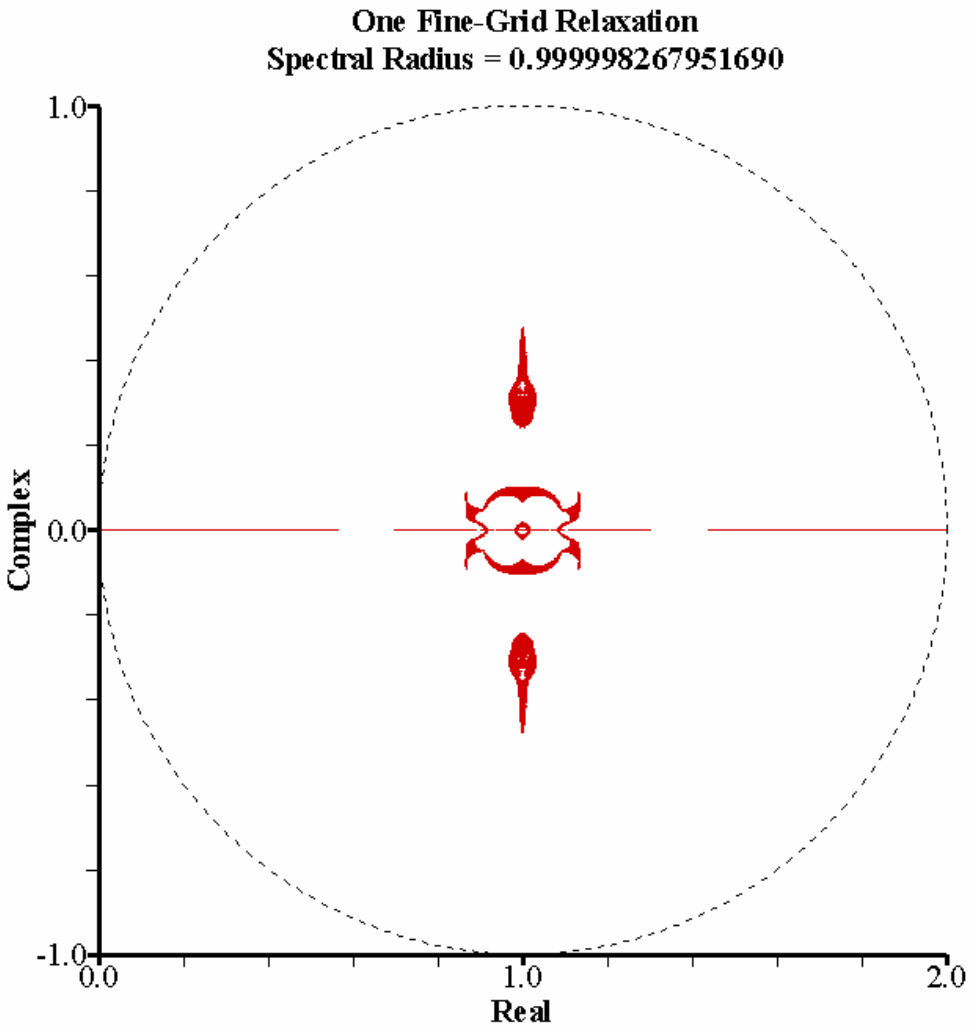

Figure 5.17 – Plot #1 for an Intermediate Problem, LBLD

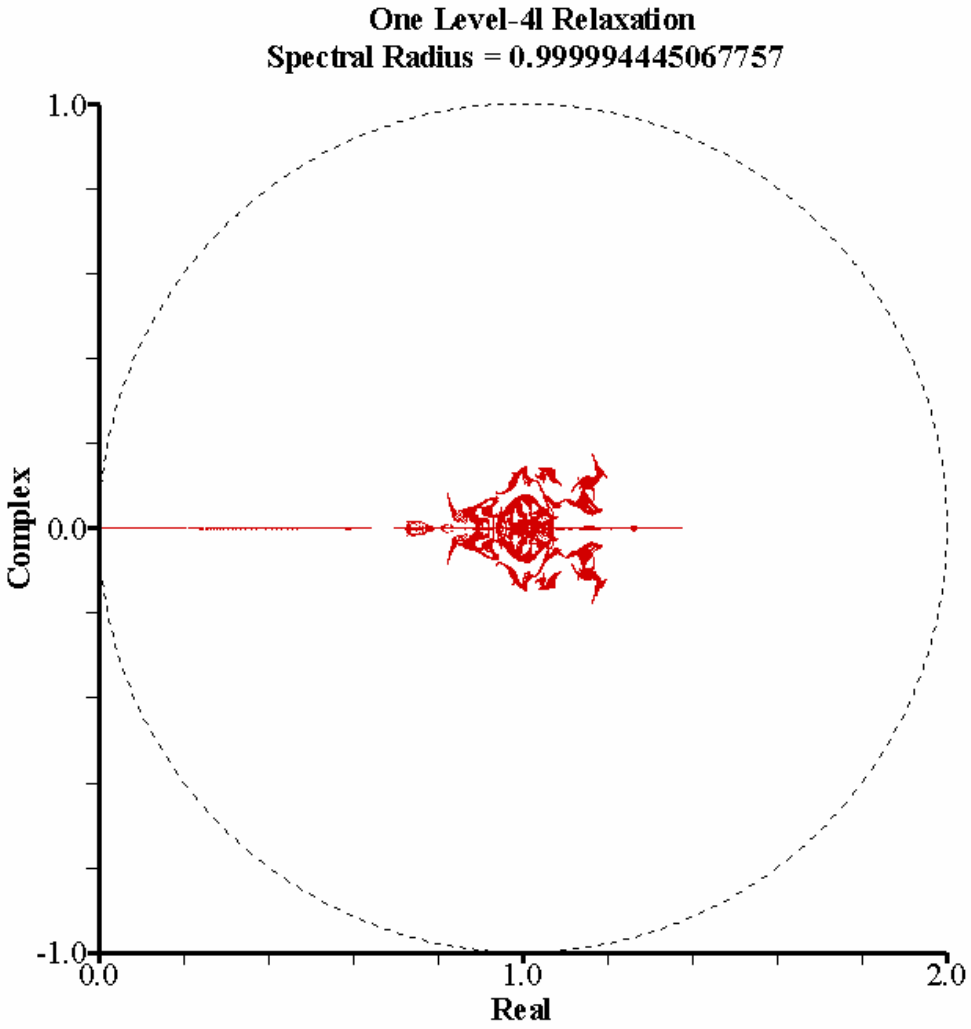

Figure 5.18 – Plot #2 for an Intermediate Problem, LBLD

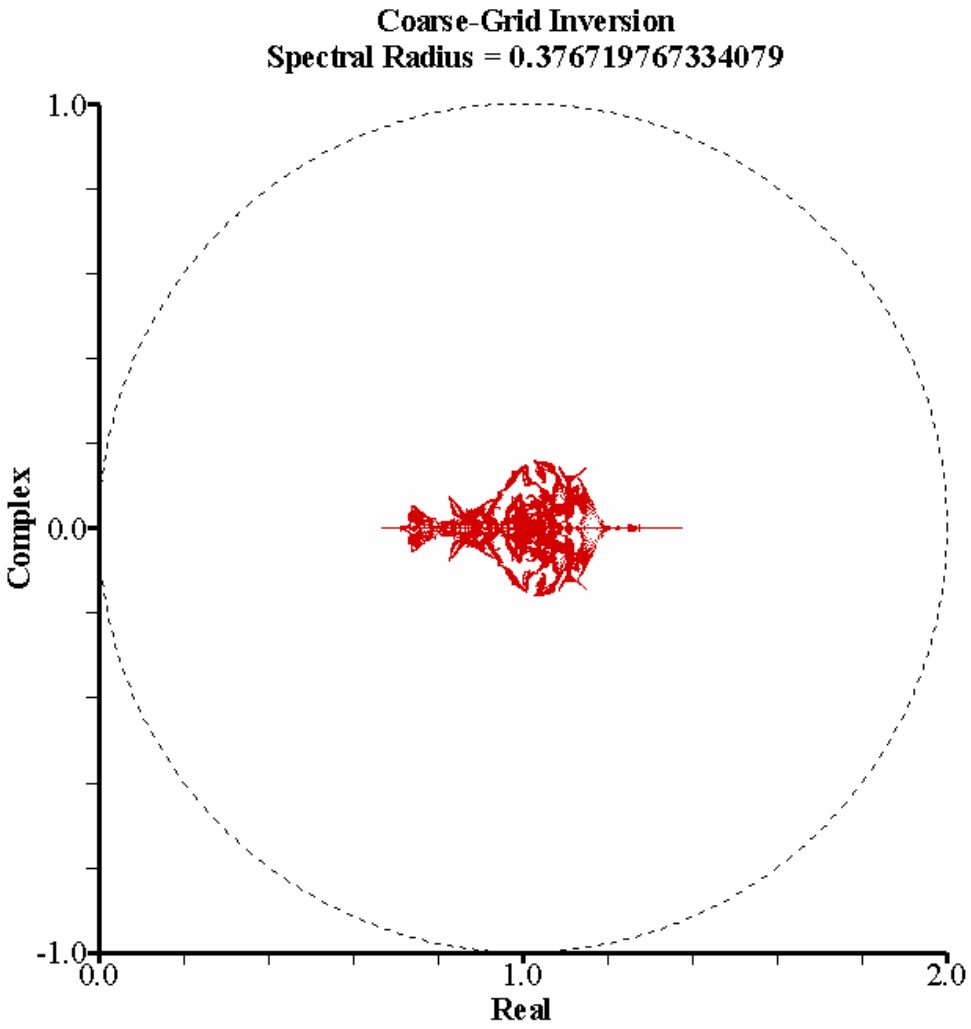

Figure 5.19 – Plot #3 for an Intermediate Problem, LBLD

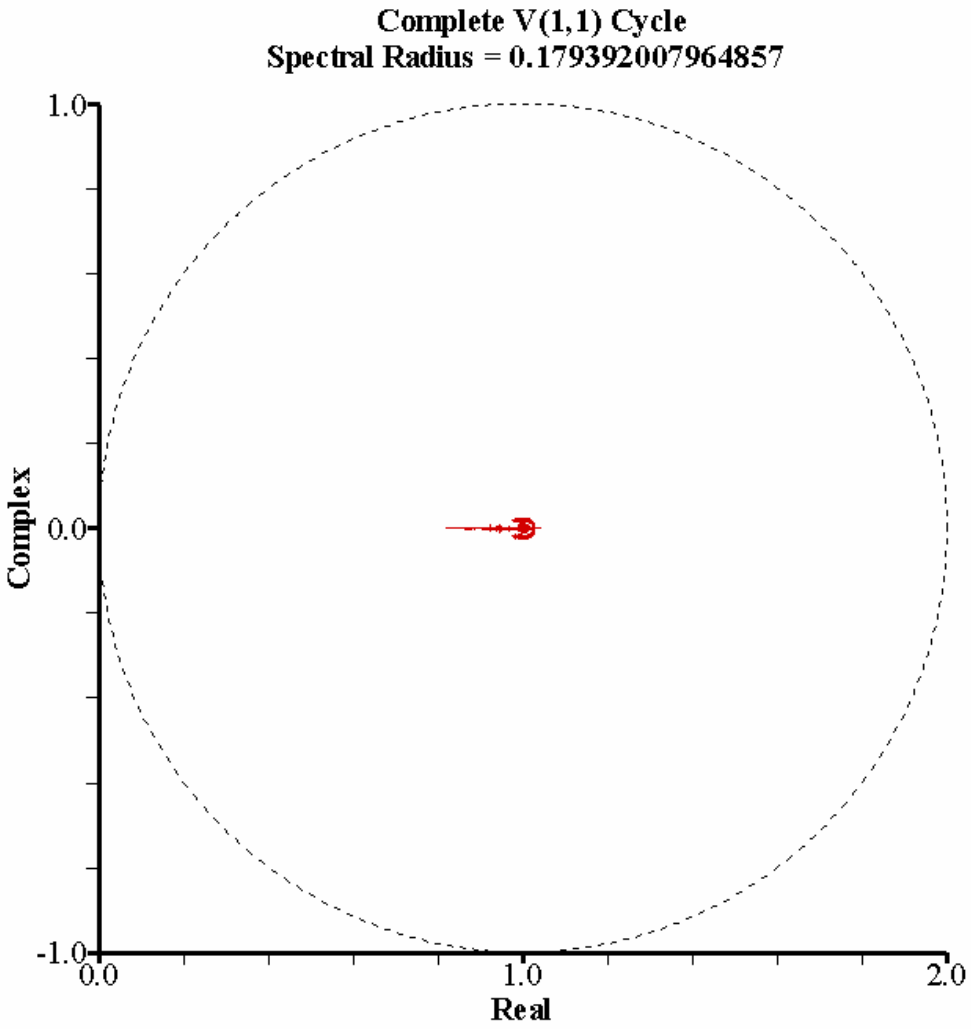

Figure 5.20 – Plot #4 for an Intermediate Problem, LBLD

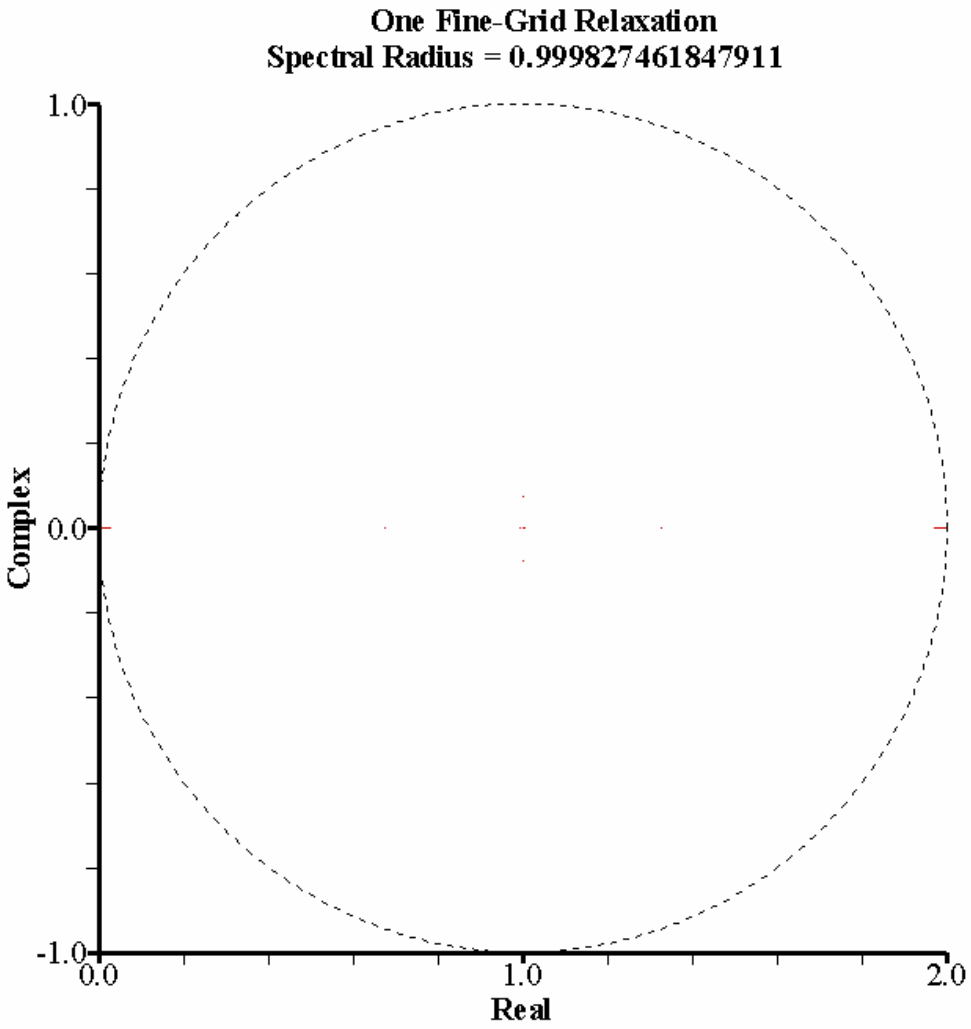

Figure 5.21 – Plot #1 for a Thick Problem, LBLD

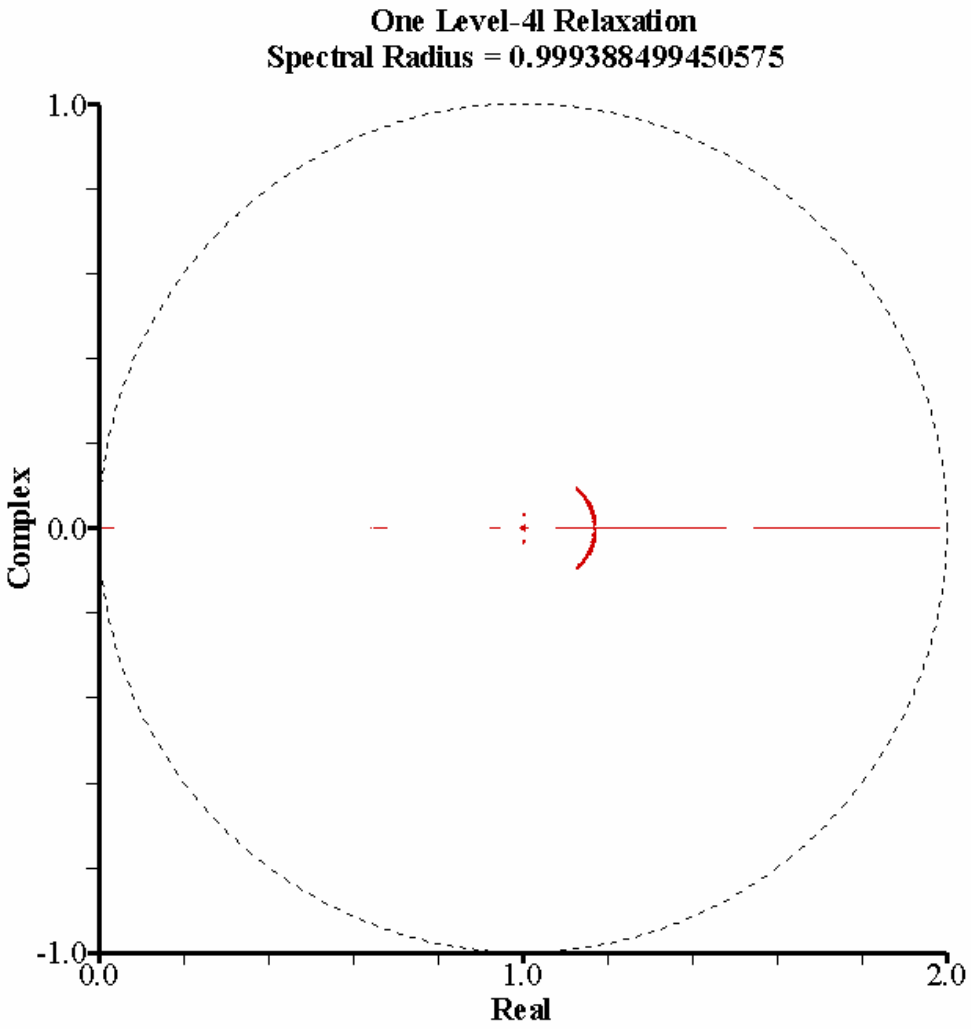

Figure 5.22 – Plot #2 for a Thick Problem, LBLD

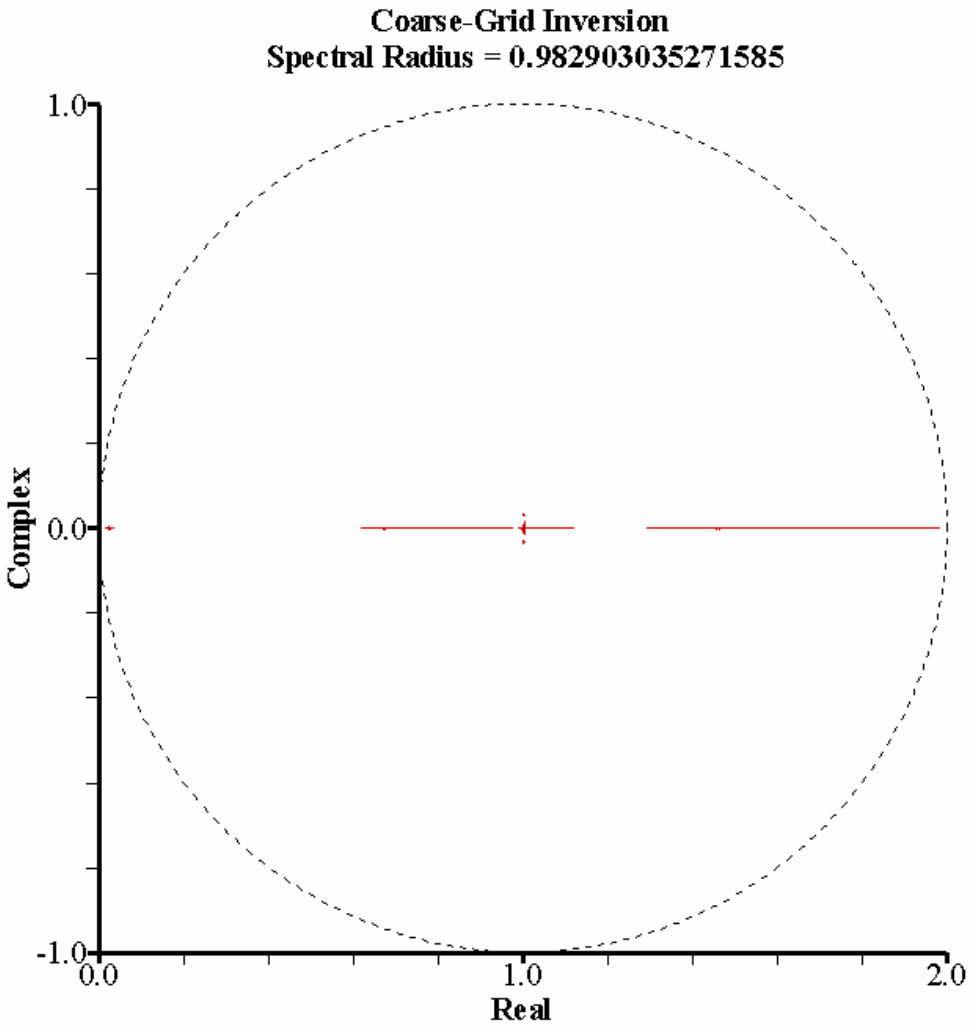

Figure 5.23 – Plot #3 for a Thick Problem, LBLD

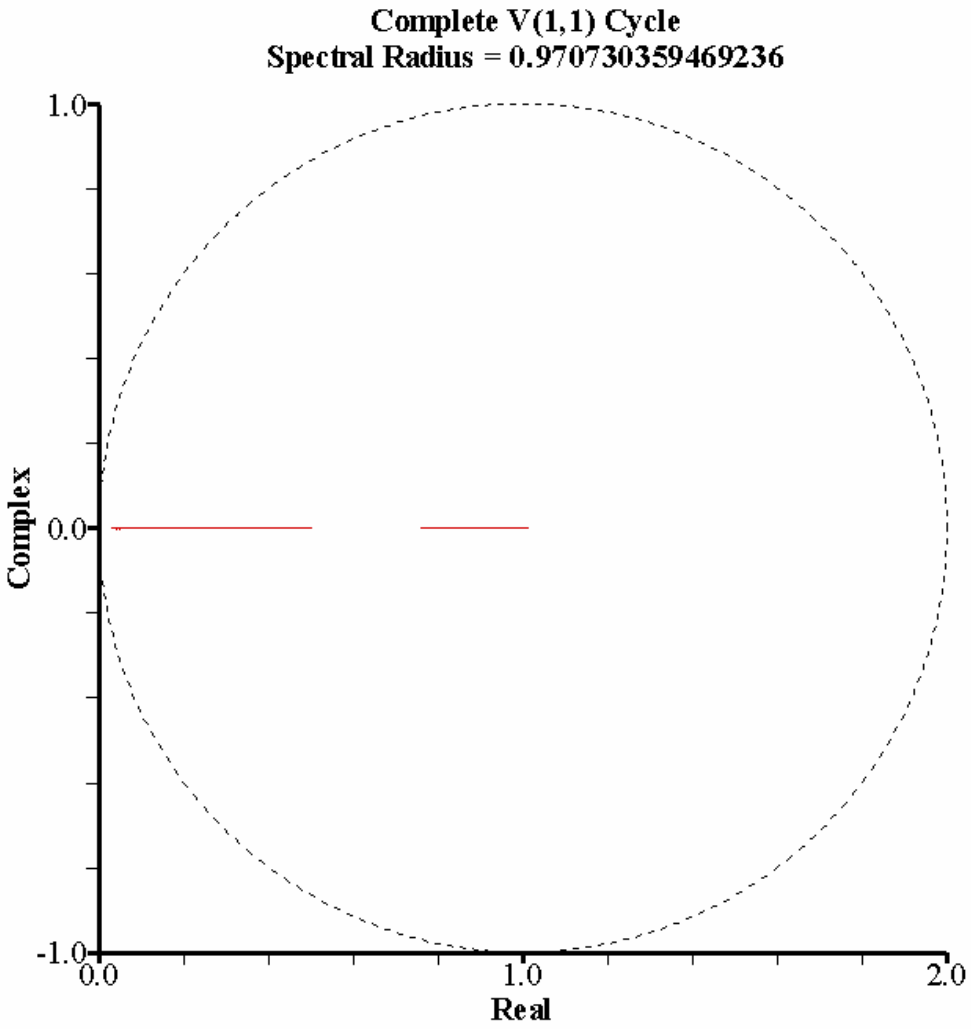

Figure 5.24 – Plot #4 for a Thick Problem, LBLD

Figures 5.13 – 5.24 show how the multigrid iteration matrix is attempting to force the individual eigenvalues to a perfect-preconditioning value of one (1). It further shows, for optically thin problems, how weakly attenuating it is on the way down the multigrid cycle which is consistent with the transport solution in thin systems. The greatest clustering of this eigenvalue spectrum occurs when the coarsest grid is solved exactly. This is consistent with our analysis of the Fourier iteration matrix. For intermediate problems, we see how inverting the coarse-grid operator eliminates error modes that cause higher convergence ratios in the general solver. For thick problems, we see that it may not be necessary to execute a complete multigrid cycle if our goal is to precondition a Krylov solver. The eigenvalues are well clustered with just simple relaxation on the fine grid.

Summary of Chapter V

In Chapter V we presented the details of implementing the Fourier iteration matrix that are specific to the two-dimensional case. Our choice of a four-cell error smoother had the unintended benefit of resulting in the same global iteration matrix in operator notation as the one-dimensional case. Individual operators are more complex in two dimensions, and there are far more error modes associated with the Fourier decomposition; however, the essential structure of the Fourier-analysis code is the same for both methods. While the global structure is the same, there are two features of the two-dimensional analysis code that distinguish it from the one-dimensional analog. First, the anticipated high wall-clock time associated with evaluating the two-dimensional iteration matrix suggests a parallel implementation. For each Fourier wave number pair $\left(\lambda_x, \lambda_y\right)$, there are four-hundred forty-eight eigenvalues and associated eigenvectors. These computations are isolated to the wave-number pair and do not require information from any other set of error modes. Consequently, the parallelization of this method is straightforward. Given a Fourier wave-number domain, we assign non-overlapping segments of that domain to different processors. Since there is no processor-to-processor communication, the time required to perform specific problems scales very well with the

number of processors. This is an observation made during the computations, but we present no timing results since they are not germane to the stated purpose of this chapter.

We test observed convergence ratios from our general two-dimensional multigrid solver, S_NAPPER_2D, against predicted results from the analysis code. Our test problem domain has problems representing each of the four problems classes discussed in the one-dimensional work. In general, our two-dimensional multigrid method is not as rapidly converging as Manteuffel's original one-dimensional method or our one-dimensional version of that method. For a homogeneous material partitioned by a grid with uniform cell spacing in both directions, we observe 1) the method performs poorly in the thin limit across a wide range of scattering ratios, lumping parameters, and quadrature orders, 2) for intermediate problems, the Fourier analysis predicts better performance than is observed with the general solver, and 3) in the thick limit, our method converges with a strong $O(\sigma h)^{-2}$ dependence on cell thickness for problems with a scattering-ratio of $c \leq 0.9999$. The first observation is the result of the slowest converging error mode being discontinuous everywhere. This is true for both methods of computing kink factors. The second observation is the result of the Fourier analysis code inverting the coarse-grid operator at grid level 16ℓ thereby eliminating all error modes that are encountered by general solver on even coarser grids. Those coarser grids have error modes that are clearly not being represented well by our bilinear finite elements resulting in poorer observed convergence ratios than those predicted by the analysis. The third observation is the result of error modes not being able to propagate through cells that are thick relative to the absorption mean-free path. Consequently, the error remaining on any four-cell problem is local and not subject to errors native to *old* information.

A final observation from this class of problems is that for purely-absorbing problems this multigrid method is an exact solver with simple relaxation on the fine grid with enough cycles to propagate information throughout the problem. The analysis code gives us results for the entire set of S_2, infinite medium problems since there is no problem boundary.

For the other three problem classes, convergence behavior and divergence mitigation techniques are observed and employed in a manner very reminiscent of the one-dimensional analog. BLD exhibits divergent behavior for highly scattering material partitioned with a non-uniform grid. This divergent behavior is eliminated by restricting kink-factors to non-negative values. This is precisely the divergent-mitigation tactic used in one dimension. Interestingly, eigenvector-based kink-factors cause the same problems to remain stable. Similar comments and conclusions can be made regarding heterogeneous, uniform problems, and heterogeneous, non-uniform problems.

All four problems classes are also used to examine this method's potential usefulness as a preconditioner for Krylov solvers. In many cases, excellent clustering of the eigenvalues is observed; however, there are cases for which this is not true. It remains an open question whether this method would be a good preconditioner for thin, highly scattering problems. These problems have error modes that are weakly attenuated by the four-cell relaxation step. Further, the slowest converging error mode can take the shape of a saddle function which is not well approximated by our bilinear finite elements. In all other cases test, the method appears to maintain the encouraging preconditioning properties observed in one dimension; however, there are interesting qualification to this statement. First, eigenvector-based kink-factors are not always the best choice. In some instances, it appears that unit-based kink-factors (which are much easier to implement in a general solver) result in better eigenvalues clustering. Second, problems can be constructed such that the divergent error mode have eigenvalues that are not well clustered outside the unit circle. This is a characteristic that was not observed in one dimension. Finally, however, the second observation is not always true.

This chapter shows the results of a successful extension of Manteuffel's original one-dimensional multigrid method to (significantly more challenging) two-dimensional problems. The final chapter, Chapter VI, summarizes the work in this dissertation and gives suggestions for future work that we believe is promising.

CHAPTER VI

CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

Summary of Results & Conclusions

While a unique solution exists to the linear Boltzmann transport equation, no analytic techniques exist for obtaining it for problems of practical interest. In practical applications the continuous problem is converted to a nested set of discrete problems in time, energy, angle, and space. The resulting large set of algebraic equations is solved iteratively. The most rapidly converging transport iterative method reported to date in the literature is a spatial multigrid method devised by Manteuffel *et al.* [27,28]. While the results these authors reported are highly encouraging, they are limited to one-dimensional problems and to a single spatial discretization, lumped linear discontinuous (LLD). The goals of our work are to understand Manteuffel's one-dimensional method, to explore how variation affect its behavior, and to extend this method to more challenging and realistic two-dimensional problems.

Our first discovery is that the one-dimensional method's performance degrades significantly in the presence of strong heterogeneities. Manteuffel *et al.* reported convergence rates that never exceeded 0.02. However, they did not test problems that were truly heterogeneous (*i.e.*, had variable scattering ratios). For strongly heterogeneous problems we have observed convergence rates as high as 0.97, and we suspect that this can be arbitrarily close to unity for arbitrarily strong heterogeneities. We conclude that the multigrid method of Manteuffel *et al.* is not as rapidly convergent as it first appeared, and perhaps not rapidly convergent enough to serve as a stand-alone iterative method for difficult transport problems, even in one dimension.

Our next result is that we were unable to find practical modifications to Manteuffel's method that render it rapidly convergent for all problems. We found that angle-dependent kink-factors did reduce the spectral radius, but not dramatically. For example, the original 0.97 became 0.91. However, angle-dependent kink-factors significantly complicate the construction and inversion of coarse-grid operators; thus,

their marginal benefit is likely outweighed by their greater cost. We reach similar conclusions about other modifications we considered.

Our basic one-dimensional algorithm behaves differently for LD than for LLD. The method generally converges somewhat slower for LD than for LLD. In addition, the LD method is susceptible to instabilities caused by negative kink-factors. The instabilities are easily eliminated by restricting kink-factors to non-negative values. We conclude that care should be taken when applying spatial multigrid methods to discrete-ordinates transport problems. Rapid convergence for one spatial discretization does not necessarily imply similarly rapid convergence for another. There are published examples of this behavior exhibited by DSA. We have shown that the same is observed with this multigrid method.

Following a multigrid cycle, the eigenvalues of the one-dimensional Fourier iteration matrix accumulate into clusters. This behavior is observed for all four problem-classes using the lumped (LLD) and standard (LD) equations. Even eigenvalues corresponding to divergent error modes for certain LD problems become clustered and are always in the right-half of the complex plane. Further, restricting kink-factors to non-negative values provides stability and does not interfere with the eigenvalues becoming tightly clustered. Krylov solvers are known to be rapidly converging if eigenvalues are tightly clustered. Thus, our one-dimensional multigrid method may be a good preconditioner for a Krylov solver.

The two-dimensional multigrid method is not as rapidly converging as the one-dimensional method, but always exhibits a smaller convergence ratio than source iteration for highly scattering problems. The Fourier analysis of the two-dimensional method provides a detailed look at the distribution of the eigenvalues at any stage of a multigrid cycle. For a range of problems, the eigenvalues are observed to become more tightly clustered at each successive stage. Thus, our two-dimensional multigrid solver may be a good preconditioner in a Krylov solver.

Time-dependent deterministic transport solvers have high memory-storage demands: they must allocate storage for two space-angle-energy-dependent arrays. Our

two-dimensional multigrid method has a fixed storage cost for describing a cell at any grid level. This storage requirement is the same for any quadrature order. Thus, although our method requires more storage that source iteration or diffusion-synthetic acceleration (DSA), it does not add significant storage requirements to typical time-dependent problems.

Suggestions for Future Work

This method should be tested as a preconditioner for Krylov solvers. The clustering of eigenvalues implies that this is an option worth investigating.

This method should be expanded to include problems with anisotropic scattering. This would complicate the Sherman-Morrison decomposition of the two-cell (1D) or four-cell (2D) transport operators. In one dimension with isotropic scattering, the largest matrix that ever had to be directly inverted was 4×4. In two dimensions, the largest matrix that was left to direct inversion was 16×16. Simply moving from isotropic (P_0) to linearly anisotropic (P_1) scattering would make the same matrices become 8×8 (1D), and 48×48 (2D). These matrices would be full.

Another interesting path to pursue would be to change the shape of the multigrid cycle. In this work, we concentrated on the $V(\nu_1,\nu_2)$ cycle. Manteuffel has shown that it works quite well for many problems and it is simple to impalement. In instances where intermediate-grid error-relaxation steps (e.g., the thin, two-dimensional problems) do little to attenuate the error, perhaps a *W*-cycle would improve convergence behavior.

Finally, it would be interesting to investigate angular coarsening in addition to spatial coarsening. The kink-factors we employ in the research are based on the scalar flux residual. It is reasonable to question whether we are getting any benefit from solving the coarse-grid problems at the finest angular resolution. A good starting point for this investigation would be our three-level Fourier analysis.

REFERENCES

[1] Case K.M., Zweifel P.F., Linear Transport Theory, Addison-Wesley Publishing Co., Reading, Massachusetts, 1967.

[2] Bell G.I., Glasstone S., Nuclear Reactor Theory, Robert E. Krieger Publishing Co., Inc., Malabar, Florida, 1970.

[3] Lewis E.E., Miller W.F., Computational Methods of Neutron Transport, American Nuclear Society, Inc., La Grange Park, Illinois, 1993.

[4] Baker R.S., Koch K.R., An S_N Algorithm for the Massively Parallel CM-200 Computer, Nuclear Science and Engineering 128 (1998) 312-320.

[5] Adams M.L., Larsen E.W., Fast Iterative Methods for Discrete-Ordinates Particle Transport Problems, Progress in Nuclear Energy 40 (1) (2002) 3-159.

[6] Trottenberg U., Oosterlee C., Schüller A., Multigrid, Academic Press, San Diego, California, 2002.

[7] Brandt A., Multi-Level Adaptive Solutions to Boundary-Value Problems, Mathematics of Computation 31 (138) (1977) 333-390.

[8] Braess D., Finite Elements Theory, Fast Solvers, and Applications in Solid Mechanics, Cambridge University Press, New York, New York, 1997.

[9] Mavriplis D.J., Multigrid Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes, AIAA Journal 26 (7) (1988) 824-831.

[10] Caughey D.A., Diagonal Implicit Multigrid Algorithm for the Euler Equations, AIAA Journal 26 (7) (1988) 841-851.

[11] Baysal O., Fouladi K., Lessard V., A Multigrid and Upwind Viscous Flow Solver on 3-D Embedded and Overlapping Grids, 27th Aerospace Sciences Meeting, Reno, Nevada, January 9-12, AIAA Paper 89-0464, 1989.

[12] Bonhaus D.L., An Upwind Multigrid Method for Solving Viscous Flows on Unstructured Triangular Meshes, Masters thesis, School of Engineering and Applied Science, George Washington University, Washington, D.C., 1993.

[13] Venkatakrishnan V., Mavriplis D.J., Agglomeration Multigrid for the Three-Dimensional Euler Equations, 32nd Aerospace Sciences Meeting & Exhibit, Reno, Nevada, January 10-13, AIAA Paper 94-0069, 1994.

[14] Parthasarathy V., Kallinderis Y., A New Multigrid Approach for 3D Unstructured, Adaptive Grids, 32nd Aerospace Sciences Meeting & Exhibit, Reno, Nevada, January 10-13, AIAA Paper 94-0078, 1994.

[15] Blazek J., A Multigrid LU-SSOR Scheme for the Solution of Hypersonic Flow Problems, 32nd Aerospace Sciences Meeting & Exhibit, Reno, Nevada, January 10-13, AIAA Paper 94-0062, 1994.

[16] Braaten M.E., Connell S.D., A 3-D Unstructured Adaptive Multigrid Scheme for the Navier-Stokes Equations, Technical Information Series, Report No. 94CRD146, GE Research & Development Center, August 1994.

[17] Anderson W.K., Rausch R.D., Bonhaus D.L., Implicit/Multigrid Algorithms for Incompressible Turbulent Flows on Unstructured Grids, AIAA Paper 95-1740, 1995.

[18] Mavriplis D.J., Viscous Flow Analysis Using a Parallel Unstructured Multigrid Solver, AIAA Journal 38 (11 (November)) (2000) 2067-2076.

[19] Mavriplis D.J., An Assessment of Linear Versus Nonlinear Multigrid Methods for Unstructured Mesh Solvers, Journal of Computational Physics 175 (1) (2002) 302-325.

[20] Nowak P.F., Larsen E.W., Martin W.R., Multigrid Methods for S_N Problems, Transactions of the American Nuclear Society 55 (1987) 355-356.

[21] Nowak P.F., Larsen E.W., Martin W.R., A Multigrid Method for S_N Calculations in X-Y Geometry, Transactions of the American Nuclear Society 56 (1988) 291-292.

[22] Barnett A., Morel J.E., Harris D.R., A Multigrid Acceleration Method for the 1D S_N Equations with Anisotropic Scattering, Nuclear Science and Engineering 102 (1989) 1-21.

[23] Manteuffel T.A., McCormick S.F., Morel J.E., Yang G., Fast Multigrid Solver for Neutron Transport Problems, in: Proceedings of the IMACS 1st International Conference on Computational Physics, Boulder, Colorado, June 11-15, 1990, 25-42.

[24] Manteuffel T.A., McCormick S.F., Morel J.E., Oliveria S., Yang G., Parallel Multigrid Methods for Transport Equations, in: Proceedings of the Copper Mountain Conference on Iterative Methods, April 10-15, 1992, 1-33.

[25] Manteuffel T.A., Ressel K., Multilevel Methods for Transport Equations in Diffusive Regimes, in: Proceedings of the Copper Mountain Conference on Multigrid Methods, April 5-9, 1993, 256-288.

[26] Manteuffel T.A., McCormick S.F., Morel J.E., Oliveria S., Yang G., A Parallel Version of a Multigrid Algorithm for Isotropic Transport Equations, SIAM Journal on Scientific and Statistical Computing 15 (1994) 474-493.

[27] Manteuffel T.A., McCormick S.F., Morel J.E., Oliveria S., Yang G., A Fast Multigrid Algorithm for Isotropic Transport Problems I: Pure Scattering, SIAM Journal on Scientific Computing 16 (1995) 601-635.

[28] Manteuffel T.A., McCormick S.F., Morel J.E., Yang G., A Fast Multigrid Algorithm for Isotropic Transport Problems II: With Absorption, SIAM Journal on Scientific Computing 17 (6) (1996) 1449-1474.

[29] Banoczi J.M., Kelley C.T., A Fast Multilevel Algorithm for the Solution of Nonlinear Systems of Conductive-Radiative Heat Transfer Equations in Two Space Dimensions, SIAM Journal on Scientific Computing 20 (1999) 1214-1228.

[30] Manteuffel T.A., Ressel K.J., Starke G., A Boundary Functional for the Least Squares Finite Element Solution of the Neutron Transport Equation, SIAM Journal on Numerical Analysis 37 (2) (2000) 556-586.

[31] Balsara D., Fast and Accurate Discrete Ordinates Methods for Multidimensional Radiative Transfer. Part I, Basic Methods, Journal of Quantitative Spectroscopy and Radiative Transfer 69 (2001) 671-707.

[32] Chang J.H., Efficient Algorithms for Discrete-Ordinates Transport Iterations in Massively Parallel Computers, Ph.D. dissertation, Department of Nuclear Engineering, Texas A&M University, College Station, 2004.

[33] Ipsen I.C.F., Meyer C.D., The Idea Behind Krylov Methods, The American Mathematical Monthly 105 (10 (December)) (1998) 889-899.

[34] Campbell S.L., Ipsen I.C.F., Kelley C.T., Meyer C.D., GMRES and the Minimal Polynomial, Bit Numerical Mathematics 36 (4) (1996) 664-675.

[35] Adams M.L., Discontinuous Finite Element Transport Solutions in Thick Diffusive Problems, Nuclear Science and Engineering 137 (2001) 298-333.

VITA

Brian David Lansrud was born in Houston, Texas on August 3, 1972. He began his education attending three Houston Independent School District (HISD) institutions – Lula M. Steven Elementary School, F.M. Black Middle School, and S.P. Waltrip High School. Following high school graduation he began a long college career at Texas A&M University in College Station, Texas. While there, he earned Bachelor of Science degrees in mathematics (August 1995) from the College of Science, and radiological health engineering (December 1995) from the College of Engineering. His graduate career began studying experimental accelerator analysis techniques and concluded studying numerical methods for solving the Boltzmann transport equation.

Brian conducted the bulk of his dissertation research with the Applied Physics (X) Division at Los Alamos National Laboratory and accepted a technical staff member position upon its completion. His research interests include advanced spatial discretization, adaptive angular techniques, and advanced preconditioning techniques for solving all forms of Boltzmann's equation. Necessarily, these research interests extend to their efficient implementation on massively parallel computing platforms. Brian can be contacted by email (bdl@lanl.gov) or by standard mail (Brian David Lansrud – MS T086; Los Alamos National Lab; P.O. Box 1663; Los Alamos, NM 87545).